TRAJECTORY OPTIMIZATION OF A MOBILE CAMERA SYSTEM FOR MAXIMIZING OPTICAL CHARACTER RECOGNITION

A Dissertation
Presented to
The Academic Faculty

By

Alexander Zabaldo

In Partial Fulfillment
of the Requirements for the Degree
Master of Science in the
Woodruff School of Mechanical Engineering

Georgia Institute of Technology

August 2021

TRAJECTORY OPTIMIZATION OF A MOBILE CAMERA SYSTEM FOR MAXIMIZING OPTICAL CHARACTER RECOGNITION

Thesis committee:

Dr. Jun Ueda, Advisor
School of Mechanical Engineering
Georgia Institute of Technology

Dr. Ai-Ping Hu
School of Mechanical Engineering
Georgia Institute of Technology

Dr. Ye Zhao
School of Mechanical Engineering
Georgia Institute of Technology

Date approved: July 13, 2021

ACKNOWLEDGMENTS

I would like to acknowledge my advisor, Dr. Jun Ueda, for being an invaluable mentor on this project as well as many previous endeavors. Learning and working under his guidance before and throughout my graduate studies has provided me an incredible opportunity to grow as a student and engineer.

I would like to thank my reading committee members, Dr. Ye Zhao and Dr. Ai-Ping Hu, for their generous time in reviewing this work and constructive feedback.

Finally, I would like to show appreciation to my fellow lab members in the Biorobotics and Human Modeling Lab: Antonio Moualeu, Mike Libretto, Waiman Meinhold, Yingxin Qiu, and Danny Martinez. Working alongside all of you has been a supportive resource that has helped me greatly throughout my time in the lab.

This project was made possible through generous funding and support in part by the National Science Foundation under Grant No. 1662029. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author and do not necessarily reflect the views of the National Science Foundation.

TABLE OF CONTENTS

Acknowledgments iii

List of Tables vii

List of Figures ix

Summary xiv

Chapter 1: Motivation and Background 1

1.1 Motivation 1

1.2 Overview 2

1.3 Research Objectives 3

Chapter 2: Literature Review 4

2.1 General Motion Deblurring 4

2.2 Non-blind Motion Deblurring 4

2.3 Motion Planning 5

2.4 Optical Character Recognition 6

2.5 Optimization Methods 6

2.6 Spectral Analysis 7

Chapter 3: Methodology 8

3.1 Problem Statement 8

3.2 Dynamics Based Motion Deblurring and Point Spread Function Determination 8

3.2.1 Image Formation 8

3.2.2 Residence Time Distribution and Point Spread Function 9

3.2.3 Non-Blind Motion Deblurring 10

3.3 Motion Planning 11

3.4 Radial Basis Function Networks 12

3.5 Optimization 13

3.5.1 Image Evaluation 15

3.5.2 Optical Character Recognition 15

3.6 Spectral Analysis 16

Chapter 4: Simulation 18

4.1 Simulation Setup 18

4.2 Simulation Results 21

4.3 Spectral Analysis 34

4.4 Discussion 43

Chapter 5: Experimentation 44

5.1 Experiment Setup 44

5.2 Experiment Results 45

5.3 Spectral Analysis 53

5.4 Discussion 60

Chapter 6: Concluding Remarks 62

6.1 Conclusion 62

6.2 Future Work 63

Appendices 64

References 69

LIST OF TABLES

4.1 Main search simulation parameters 20

4.2 Main search simulation score means μ_{OCR}, standard deviations σ_{OCR}, and t-test scores t_{OCR} by trajectory over five trials 26

4.3 Main search simulation t-test p-values matrix by trajectory 27

4.4 Acceleration limit simulation score means μ_{OCR}, standard deviations σ_{OCR}, and t-test scores t_{OCR} by trajectory over five trials 30

4.5 Acceleration limit simulation t-test p-values matrix by trajectory 30

4.6 Exposure time simulation score means μ_{OCR}, standard deviations σ_{OCR}, and t-test scores t_{OCR} by trajectory over five trials 33

4.7 Exposure time simulation t-test p-values matrix by trajectory 33

4.8 Main search simulated power spectra difference means μ_{PS}, standard deviations σ_{PS}, and t-test scores t_{PS} by trajectory over five trials 37

4.9 Main search simulation power spectra score t-test p-values matrix by trajectory 38

4.10 Acceleration limit simulated power spectra difference means μ_{PS}, standard deviations σ_{PS}, and t-test scores t_{PS} by trajectory over five trials 40

4.11 Acceleration limit simulation power spectra score t-test p-values matrix by trajectory 40

4.12 Exposure time simulated power spectra difference means μ_{PS}, standard deviations σ_{PS}, and t-test scores t_{PS} by trajectory over five trials 42

4.13 Exposure time simulation power spectra score t-test p-values matrix by trajectory 43

5.1 Main search experiment recorded vs. attempted normalized cross correlation X with t-test scores vs. all attempted t_{att} and t-test scores vs. all recorded t_{rec} 49

5.2 Main search experiment score means μ_{OCR}, standard deviations σ_{OCR}, and t-test scores t_{OCR} by trajectory 49

5.3 Main search experiment t-test p-values matrix by trajectory 50

5.4 Acceleration limit experiment recorded vs. attempted normalized cross correlation X with t-test scores vs. all attempted t_{att} and t-test scores vs. all recorded t_{rec} 52

5.5 Acceleration limit experiment score means μ_{OCR}, standard deviations σ_{OCR}, and t-test scores t_{OCR} by trajectory 53

5.6 Acceleration limit experiment t-test p-values matrix by trajectory 53

5.7 Main search experiment power spectra difference means μ_{PS}, standard deviations σ_{PS}, and t-test scores t_{PS} by trajectory 57

5.8 Main search experiment power spectra score t-test p-values matrix by trajectory 57

5.9 Acceleration limit experiment power spectra difference means μ_{PS}, standard deviations σ_{PS}, and t-test scores t_{PS} by trajectory 59

5.10 Acceleration limit experiment power spectra score t-test p-values matrix by trajectory 60

A1 Image set ground truth text and mean static OCR score 66

A2 Sample simulation scores for RBF4 with trial 1 OCR text by image 67

A3 Sample experiment scores for RBF4 with OCR text by image 68

LIST OF FIGURES

1.1 Vehicle application of a mobile camera platform for street sign text recognition 2

3.1 A visual representation of the components that comprise a captured image . 9

3.2 Residence time distribution across a non-stationary trajectory with exposure position x_e 10

3.3 Example of an image deblurred with Richardson-Lucy deconvolution with strong ringing artifacts 11

3.4 An example five function symmetric RBF network with $\epsilon = 3$ and individual functions shown 13

3.5 Power spectra of a static and deblurred image and the formation of the difference power spectrum with separated central horizontal third 17

4.1 Flowchart of the simulation's optimization algorithm 19

4.2 Output plots of the optimization search including individual and set scores as well as trajectories attempted 21

4.3 Sample of simulated deblurred image using linear trajectory with position, velocity, and residence time distribution plots 22

4.4 Sample of simulated deblurred image using inverse error trajectory with position, velocity, and residence time distribution plots 22

4.5 Sample of simulated deblurred image using RBF1 trajectory with position, velocity, and residence time distribution plots 23

4.6 Sample of simulated deblurred image using RBF2 trajectory with position, velocity, and residence time distribution plots 23

4.7 Sample of simulated deblurred image using RBF3 trajectory with position, velocity, and residence time distribution plots 24

4.8 Sample of simulated deblurred image using RBF4 trajectory with position, velocity, and residence time distribution plots 24

4.9 Sample of simulated deblurred image using RBF5 trajectory with position, velocity, and residence time distribution plots 25

4.10 Sample of simulated deblurred image using an optimized trajectory for a $1g$ acceleration limit with position, velocity, and residence time distribution plots 27

4.11 Sample of simulated deblurred image using an optimized trajectory for a $2g$ acceleration limit with position, velocity, and residence time distribution plots 28

4.12 Sample of simulated deblurred image using an optimized trajectory for a $3g$ acceleration limit with position, velocity, and residence time distribution plots 28

4.13 Sample of simulated deblurred image using an optimized trajectory for a $4g$ acceleration limit with position, velocity, and residence time distribution plots 29

4.14 Sample of simulated deblurred image using an optimized trajectory for a a 1/15 second exposure time with position, velocity, and residence time distribution plots 31

4.15 Sample of simulated deblurred image using an optimized trajectory for a a 1/30 second exposure time with position, velocity, and residence time distribution plots 31

4.16 Sample of simulated deblurred image using an optimized trajectory for a a 1/45 second exposure time with position, velocity, and residence time distribution plots 32

4.17 Sample of simulated deblurred image using an optimized trajectory for a a 1/60 second exposure time with position, velocity, and residence time distribution plots 32

4.18 Static image 1 and its power spectra 34

4.19 Deblurred image power spectra from simulated linear trajectory and power spectra difference from static image power spectra 34

4.20 Deblurred image power spectra from simulated inverse error trajectory and power spectra difference from static image power spectra 35

4.21 Deblurred image power spectra from simulated RBF1 and power spectra difference from static image power spectra 35

4.22 Deblurred image power spectra from simulated RBF2 and power spectra difference from static image power spectra 35

4.23 Deblurred image power spectra from simulated RBF3 and power spectra difference from static image power spectra 36

4.24 Deblurred image power spectra from simulated RBF4 and power spectra difference from static image power spectra 36

4.25 Deblurred image power spectra from simulated RBF5 and power spectra difference from static image power spectra 36

4.26 Deblurred image power spectra from simulated $1g$ acceleration limit trajectory and power spectra difference from static image power spectra 38

4.27 Deblurred image power spectra from simulated $2g$ acceleration limit trajectory and power spectra difference from static image power spectra 39

4.28 Deblurred image power spectra from simulated $3g$ acceleration limit trajectory and power spectra difference from static image power spectra 39

4.29 Deblurred image power spectra from simulated $4g$ acceleration limit trajectory and power spectra difference from static image power spectra 39

4.30 Deblurred image power spectra from simulated 1/15 second exposure time trajectory and power spectra difference from static image power spectra . . 41

4.31 Deblurred image power spectra from simulated 1/30 second exposure time trajectory and power spectra difference from static image power spectra . . 41

4.32 Deblurred image power spectra from simulated 1/45 second exposure time trajectory and power spectra difference from static image power spectra . . 42

4.33 Deblurred image power spectra from simulated 1/60 second exposure time trajectory and power spectra difference from static image power spectra . . 42

5.1 Experimental setup for the controlled-blur capture process 45

5.2 Sample of experiment deblurred image using linear trajectory with position, velocity, and residence time distribution plots 46

5.3 Sample of experiment deblurred image using inverse error trajectory with position, velocity, and residence time distribution plots 46

5.4 Sample of experiment deblurred image using RBF1 trajectory with position, velocity, and residence time distribution plots 46

5.5 Sample of experiment deblurred image using RBF2 trajectory with position, velocity, and residence time distribution plots 47

5.6 Sample of experiment deblurred image using RBF3 trajectory with position, velocity, and residence time distribution plots 47

5.7 Sample of experiment deblurred image using RBF4 trajectory with position, velocity, and residence time distribution plots 47

5.8 Sample of experiment deblurred image using RBF5 trajectory with position, velocity, and residence time distribution plots 48

5.9 Sample of experiment deblurred image using an optimized trajectory for a $1g$ acceleration limit with position, velocity, and residence time distribution plots 50

5.10 Sample of experiment deblurred image using an optimized trajectory for a $2g$ acceleration limit with position, velocity, and residence time distribution plots 51

5.11 Sample of experiment deblurred image using an optimized trajectory for a $3g$ acceleration limit with position, velocity, and residence time distribution plots 51

5.12 Sample of experiment deblurred image using an optimized trajectory for a $4g$ acceleration limit with position, velocity, and residence time distribution plots 51

5.13 Static experiment image 1 and its power spectra 54

5.14 Deblurred image power spectra from experiment linear trajectory and power spectra difference from static image power spectra 54

5.15 Deblurred image power spectra from experiment inverse error trajectory and power spectra difference from static image power spectra 54

5.16 Deblurred image power spectra from experiment RBF1 and power spectra difference from static image power spectra 55

5.17 Deblurred image power spectra from experiment RBF2 and power spectra difference from static image power spectra 55

5.18 Deblurred image power spectra from experiment RBF3 and power spectra difference from static image power spectra 55

5.19 Deblurred image power spectra from experiment RBF4 and power spectra difference from static image power spectra 56

5.20 Deblurred image power spectra from experiment RBF5 and power spectra difference from static image power spectra 56

5.21 Deblurred image power spectra from experiment $1g$ acceleration limit trajectory and power spectra difference from static image power spectra 58

5.22 Deblurred image power spectra from experiment $2g$ acceleration limit trajectory and power spectra difference from static image power spectra 58

5.23 Deblurred image power spectra from experiment $3g$ acceleration limit trajectory and power spectra difference from static image power spectra 58

5.24 Deblurred image power spectra from experiment $4g$ acceleration limit trajectory and power spectra difference from static image power spectra 59

A1 Image set used by the optimization consisting of score filtered images from the ICDAR 2015 competition and Street View Text Dataset 65

SUMMARY

Camera systems in motion are subject to significant blurring effects that lead to a loss of information during the image capture. This is especially damaging for optical character recognition for which edge preservation is critical to achieving a high recognition rate. Using non-blind motion deblurring, a trajectory and point spread function can be designed to maximize the recognition rate while meeting endpoint constraints. Optimization through the use of radial basis function networks can therefore be used as a way to find ideal trajectories to reduce blurring effects and preserve text sharpness. This work investigates this problem using simulation of a blurred image capture process. The simulation is automated using radial basis function network optimization and a genetic algorithm to determine trajectories with the best recognition rate. Optimized trajectories yielded recognition scores with up to 57.3% improvement in simulation compared to an analogous linear profile. These results were then verified through physical experimentation with a real-world, controlled-blur image capture process that yielded up to 29.4% improvement across the same comparison. Results were then analyzed using spectral analysis to understand why the chosen trajectories preserve text edges. These findings can be applied to a wide variety of controlled mobile camera platforms, such as autonomous automobiles or unmanned aerial vehicles, to improve their ability to gather information from their environment.

CHAPTER 1
MOTIVATION AND BACKGROUND

1.1 Motivation

Cameras allow systems to extract visual information from their environment but can be subject to many types of error and degradation. The proliferation of autonomous vehicles and other systems has only increased the need for improvement in the retrieval of relevant data from natural scenes [1]. During operation, many systems cannot simply stop and wait for a stable capture in order to receive additional environmental information, but capture in motion can lead to blurring that renders such information essentially useless. Motion blur can occur either as movement of the camera or movement of objects in the scene. Due to the nature of a limited exposure window, motion blur is a result of light being captured across multiple photosensitive elements. In some cases, the trajectory of the camera may be known due to its positioning on a controlled mobile platform, such as an autonomous vehicle or unmanned aerial vehicle. Knowledge of the camera movement allows for non-blind motion deblurring, which can, under the correct assumptions and conditions, yield a final image with high accuracy compared to the ground truth image [2].

One particular problem that has become more prevalent is the need to read street signs using machine vision for the purpose of autonomous vehicles as shown in Figure 1.1. In cases such as these, motion can be estimated as one-dimensional due to the velocity in the vertical direction relative to the horizontal being negligible. Text recognition is a unique problem in regards to motion deblurring as it has an emphasis on strong edge preservation as opposed to noise reduction seen in general motion deblurring. Specifically, the development of a motion deblurring optimization tailored specifically to text recognition would lead to improvement in the readability of natural scene text, such as street and road signs,

while in motion. Therefore, the goal of this work is to develop an algorithm to determine the ideal trajectory type for planned motion text-deblurring as well as identify these trajectories and why they yield positive results

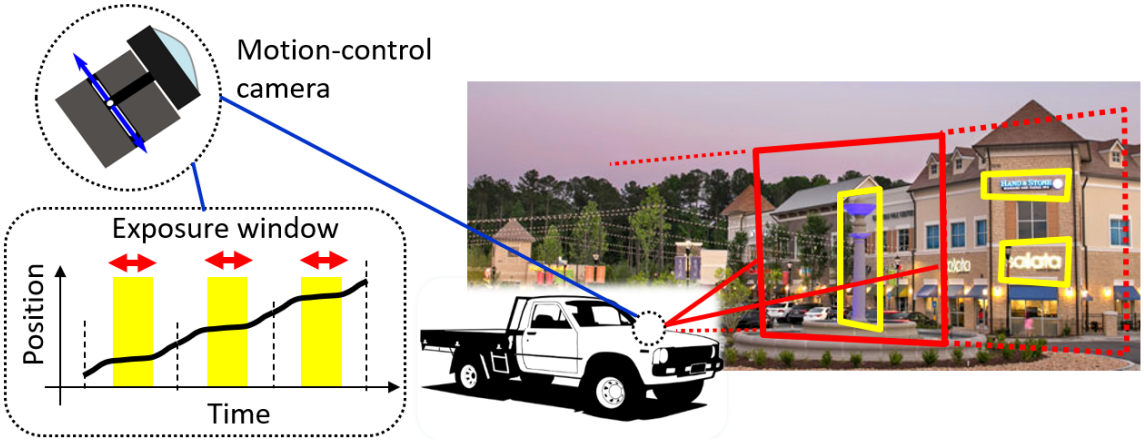

Figure 1.1: Vehicle application of a mobile camera platform for street sign text recognition

1.2 Overview

This work will first identify common issues and gaps in the existing research to show the need for text recognition-based path planning optimization. Using a combination of traditional non-blind deblurring procedures and the use of radial basis function network-based path planning optimization, an algorithm was developed in order to automatically cycle through endpoint constrained trajectories with direct feedback based on text recognition scores. A simulation was created to first find trajectories based on a simulated capture and blur system; the results of these studies, including their trajectory shapes and final images, were analyzed using methods such as spectral analysis to understand why they have the greatest reduction on blurring effects. The found trajectories were then verified using a linear stage mounted camera to induce blur in a controlled manner during a real capture process. Tests in both the simulated and physical environment were completed to assess overall best performance, the relationship between allowable acceleration and score, and the exposure time and score.

1.3 Research Objectives

This work seeks to achieve the following goals:

- Use the existing understanding of non-blind motion deblurring and radial basis function network motion planning to develop an algorithm that can automatically determine the best trajectory given endpoint constraints for text recognition in natural images
- Apply this algorithm in both simulated and real world experiments to find general trends of ideal solutions such as trajectory shape under a variety of initial settings and conditions
- Analyze the results of experimentation using spectral analysis in order to determine why the trajectories found yield the best results in text recognition

CHAPTER 2
LITERATURE REVIEW

2.1 General Motion Deblurring

Image deblurring is the process of taking a blurred image and restoring it as best as possible to a ground truth image through the removal of blurring artifacts. These artifacts are typically the result of movement of the camera within the exposure window that results in blur in the direction of the movement. There are two well-established types of motion deblurring: blind deblurring and non-blind deblurring [3]. Blind motion deblurring is an approach where the conditions of the motion blurring are unknown and generally concerns methods of estimating the motion in order to then deblur the image [4, 5]. On the other hand, non-blind motion deblurring focuses on deblurring the image as best as possible given the motion without introducing its own artifacts [6]. There is also increasing investigation in model and machine learning based deblurring techniques [7]. There has been much research conducted on motion deblurring with a variety of goals. For example, increased interest in autonomous vehicles has created a need to read text from road signs [1, 8]. Deblurring for object recognition generally seeks to restore an image with relatively low noise [9, 10, 11]. Text recognition, as is relevant for this study, depends less on image noise and significantly more on edge preservation in order for text features to be extracted by text recognition engines [12].

2.2 Non-blind Motion Deblurring

Non-blind motion deblurring is an approach to motion deblurring based upon a given assumption that either the blur kernel or information directly relating to the blur kernel is known. Often, the information is in the form of inertial data [13] or planned motion tra-

jectory information [14]. Katoch [15] developed a framework relating one-dimensional trajectory information to the blur kernel through residence time distribution. Ultimately, there are multiple methods for obtaining the blur kernel, but the deblurring process must still be applied. Two common methods for non-blind motion deblurring include Wiener deconvolution and Richardson-Lucy deconvolution [16]. Wiener deconvolution is used for real-time, rapid processing of images as it is a single deconvolution operation applied to the whole of the image [17]. On the other hand, Richardson-Lucy deblurring is an iterative method that yields significantly better results in image accuracy compared to Wiener deblurring at the expense of greater computational time [18]. Non-blind motion deblurring success is highly dependent on the trajectory that creates the blur kernel as well as how accurate the blur kernel is and a variety of approaches have been used to improve overall results [19]. One way to improve the quality of deblurred results is to apply motion planning when possible to an image capture in order to ensure blur kernel accuracy as well as design the blur kernel so as to create blurring artifacts that are easy to negate [2].

2.3 Motion Planning

Motion planning is critical in non-blind motion deblurring as the design of the camera trajectory directly relates to the formation of the blur kernel. The known previous work in dynamics-based motion deblurring utilized linear, inverse error, and fourth order polynomial functions as proposed trajectories for improving image sharpness [12]. However, these functions are limited in shape variation given position and velocity endpoint constraints. Linear and inverse error functions with constrained endpoints only have a single possible trajectory, and polynomial functions can change coefficients, but are limited in function variation at specific points in the trajectory. One way to increase the complexity of potential trajectories is to use radial basis function networks while maintaining endpoint constraints. Radial basis function network optimization [20, 21] is a well-known method for using discretized, finite, parameters to generate paths with predictable shape based on

its inputs.

2.4 Optical Character Recognition

Optical Character Recognition (OCR) is the process of extracting machine text from image text. Most modern OCR engines are machine learning based, such as Google's open source Tesseract used in this work [22]. There is a wide range of research in OCR including improvements to model based approaches as well as various applications [23]. One particular problem relevant to this research's main goals is the use of OCR in natural images containing text, such as street signs and storefronts. OCR engines are often designed for or best used on document text with significantly diminished results when applied to these natural scenes. Unique solutions such as preprocessing or temporal fusion have been used in order to improve accuracy [24]. Binarization, blob detection, and image segmentation are all methods used in order to prepare image text for feature extraction [25]. OCR also dramatically deteriorates under the effect of motion blur due to the loss of text features [26]. This is a unique challenge that can benefit from OCR-based optimization.

2.5 Optimization Methods

Optimization is widely used in both motion deblurring and optical character recognition. There are many existing, well-tested optimization methods, such as the genetic algorithm used in this work [27, 28]. This algorithm works well for finding solutions to large, non-linear problem formulations and has been used previously in optimizing non-blind motion deblurring [29]. For problems such as this, establishing a cost function is critical; various evaluation methods can be used depending on the specific application of the completed optimization. Variation from a ground truth image [30], image sharpness [31], string edit distance [32], and OCR word confidence [33] are all potential, previously described scoring methods for optimizing motion deblurring for text recognition.

2.6 Spectral Analysis

Spectral analysis is a method of evaluating signals by converting them from the time domain to the frequency domain using a Fourier Transform. Two-dimensional spectral analysis is a well established method of observing images both qualitatively and quantitatively [34]. Specifically, the frequency domain is useful for determining in what direction and to what extent blur has been imposed on an image [35]. This is especially useful for this case of text detection as edge preservation is critically important, and edges are represented as high frequency signals in the frequency domain perpendicular to their direction in the image plane [36]. Various averaging and scoring methods can be directly applied to the two-dimensional transform for objective measurement [37]. Comparison of the frequency domain images before and after some process, such as blurring and deblurring, can be especially useful as it allows for viewing of what edge information has been lost in the image [38].

CHAPTER 3

METHODOLOGY

3.1 Problem Statement

The goal of this work is to establish an optimal camera trajectory that, compared to other trajectories with the same motion and endpoint constraints, can be used to plan the motion of a mobile camera that will allow for the greatest rate of correct text recognition in natural images after image reconstruction. Trajectories are generated using radial basis function networks in a one-dimensional space and subject to optimization via a cost function based on the incorrect text recognition rate. Numerical optimization is used to attempt motion deblurring in an automated fashion against a library of text-based natural images.

3.2 Dynamics Based Motion Deblurring and Point Spread Function Determination

3.2.1 Image Formation

Images are formed based on the intensity of light exposed onto the photosensitive elements of the camera sensor. During this image capture process, two types of noise are introduced: shot noise and thermal noise [12]. Shot noise is the result of variance in the number of photons received by each photosensitive element over time due to changes in the scene or changes in the camera position $\boldsymbol{x}(t)$. Thermal noise $\boldsymbol{N}$ is created within the camera sensor via variation in the electrical signal due to changes in temperature and is modeled as an additive zero-mean Gaussian $\mathscr{N}$ with variance σ^2. Shot noise can be modeled as a blurry image $\boldsymbol{B}$ generated by a stationary Poisson process $\mathscr{P}$ with intensity λ:

$$\boldsymbol{B} \sim \mathscr{P}\left(\lambda \int_{T_e} \boldsymbol{L}(\boldsymbol{x}(t))dt\right), \boldsymbol{N} \sim \mathscr{N}(0, \sigma^2), \tag{3.1}$$

where $\boldsymbol{L}$ is the latent image. The captured image $\boldsymbol{I}$ can then be defined as:

$$\boldsymbol{I} = \boldsymbol{B}(\boldsymbol{x}(t), T_e, \lambda) + \boldsymbol{N}, \tag{3.2}$$

or as:

$$\boldsymbol{I} = \boldsymbol{K} \otimes \boldsymbol{L} + \boldsymbol{N}, \tag{3.3}$$

where T_e is the exposure window, $\boldsymbol{K}$ is the blur kernel or Point Spread Function (PSF) and $\otimes$ is the convolution operator. A visual representation of the capture process is shown in Figure 3.1.

Figure 3.1: A visual representation of the components that comprise a captured image

3.2.2 Residence Time Distribution and Point Spread Function

Knowledge of the camera's motion is critical as it is needed to generate a Residence Time Distribution (RTD). An RTD describes the amount of time the camera spends over a certain section of the image [15]. These image sections evenly divide the image based upon a set step size, image capture exposure time, and linear capture distance. The RTD is represented graphically in Figure 3.2 and in the equation below:

$$\tau(\boldsymbol{x}) = \frac{1}{\|\dot{\boldsymbol{x}}(\boldsymbol{x})\|_2}, \dot{\boldsymbol{x}}(\boldsymbol{x}) \neq [0, 0], \tag{3.4}$$

where $\dot{\boldsymbol{x}}(\boldsymbol{x})$ is the velocity of the camera at a given point over the image and $\tau(\boldsymbol{x})$ is the RTD. In the case that the camera is stationary, the result is a Dirac delta function centered on the trajectory center point with a value of the image exposure time. The sum of the RTD

is equal to the total exposure time ΔT for any given trajectory.

Using the RTD found via the known trajectory, the PSF can be determined. This PSF is used as the blur kernel for any deconvolution method as it describes the light energy distribution for a pixel in a captured image. For a one dimensional case, the PSF is represented by the RTD normalized and equally sampled to the size of the blur kernel imposed on the center row with all other elements equal to zero. The size of the blur kernel is determined by scaling the image width in pixels to the ratio of the image capture width and the displacement traveled during capture.

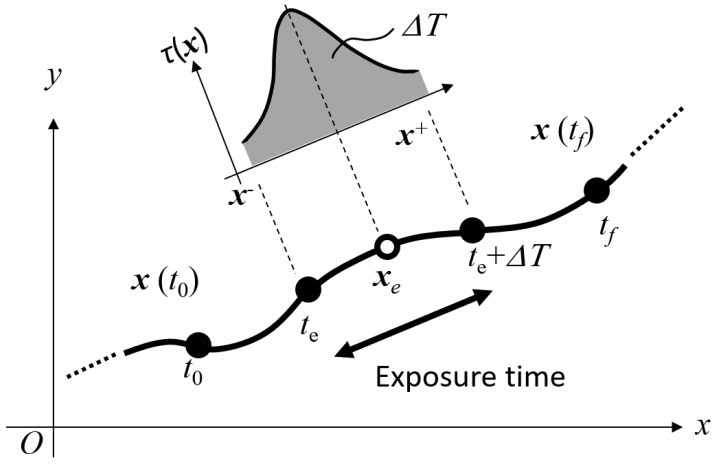

Figure 3.2: Residence time distribution across a non-stationary trajectory with exposure position x_e

3.2.3 Non-Blind Motion Deblurring

The reconstruction of a blurred image via non-blind deblurring is based upon the formation of the captured image described in Equation 3.3. Richardson-Lucy deconvolution is a well-known iterative approach that is computationally slower and more intensive than other methods, but produces more accurate results and is therefore used here. It can be modeled as:

$$\boldsymbol{I}^{j+1} = (\boldsymbol{B} \oslash (\boldsymbol{I}^{j} \otimes \boldsymbol{K}) \otimes \overline{\boldsymbol{K}}) \odot \boldsymbol{I}^{j}, \tag{3.5}$$

where $\boldsymbol{I}^{j}$ is the image estimate at the j-th iteration, $\overline{\boldsymbol{K}}$ is the blur kernel flipped along

the central horizontal and central vertical axes, $\odot$ is the Hadamard product, and $\oslash$ is the Hadamard divide [39]. In a one-dimensional case, the $\overline{\boldsymbol{K}}$ matrix is the same as $\boldsymbol{K}$ as this matrix is a single row of symmetric values along the center row with all other values equal to zero. While Richardson-Lucy deconvolution generally performs better than other common methods, it generally cannot perfectly recreate images even with a known blur kernel. It can also introduce "ringing" artifacts, shown as bands near the edges of the image in Figure 3.3 that can be detrimental to procedures like text recognition.

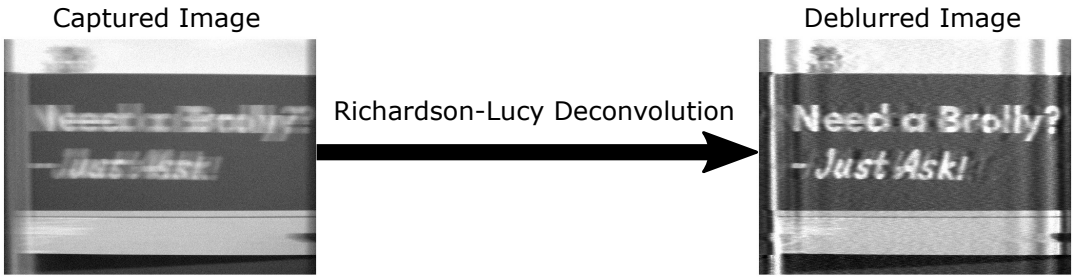

Figure 3.3: Example of an image deblurred with Richardson-Lucy deconvolution with strong ringing artifacts

3.3 Motion Planning

Motion planning for non-blind motion deblurring describes the design of a blur kernel via control of an RTD through the trajectory profile. For the purposes of this work, it is assumed that the velocity in the vertical direction is negligible relative to the velocity in the horizontal direction, establishing one-dimensional motion. For a camera in motion, an exposure position x_e is established as the average or center position during the entire exposure window T_e:

$$x_e = \frac{1}{\Delta T} \int_{T_e} x(t) dt, \tag{3.6}$$

There are several limitations imposed on the designed trajectory including that it is one-dimensional, is symmetric centered upon x_e, must be completed within ΔT, and must meet position and velocity endpoint constraints. In order to meet these conditions, this thesis proposes the implementation of radial basis function networks for path planning.

3.4 Radial Basis Function Networks

A radial basis function (RBF) is any real valued function in which the value is solely dependent on the difference between the input and some given center, defined as $\varphi(x) = \varphi(||x - c||)$ where x is the primary function variable and c is the RBF center [20]. The most common RBF used is the Gaussian function:

$$\varphi(r) = e^{-(\epsilon r)^2}, \tag{3.7}$$

where r is the primary function variable and ϵ is the Gaussian shape factor. The shape factor is directly related to the standard deviation of the Gaussian and therefore controls each function's spread around a particular center. It is typically fixed for a given set of RBFs. Radial basis function networks use multiple RBFs and an equal number of weights and function offsets to generate a weighted sum of the functions:

$$s(t) = \sum_{k=0}^{n} w_k \varphi(||t - t_k||), \tag{3.8}$$

where $s(t)$ is the RBF network function, n is the number of RBF functions used, w_k is the function weight, t_k is the function offset, and t is the trajectory timescale. An example of the formation of an RBF network is shown in Figure 3.4.

One advantage of RBF networks is the ability to easily and predictably vary their shape using the weight and offset values [21]. Given an RBF network of n RBF functions, parameters to fully describe the network can be formed as a series of n $[w_k, t_k]$ pairs. This set of parameters can be used as inputs for a number of optimization or search methods to determine optimal sets with a given cost function. It is also known that any combination of values is guaranteed to produce a smooth function.

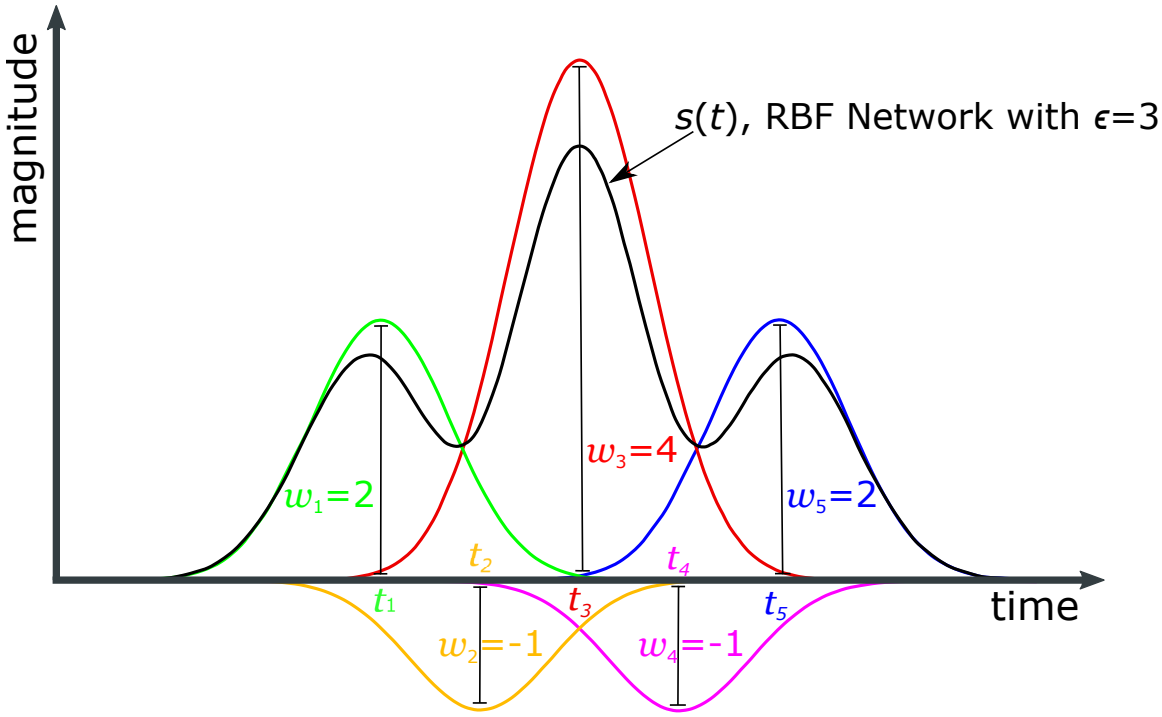

Figure 3.4: An example five function symmetric RBF network with $\epsilon = 3$ and individual functions shown

3.5 Optimization

Proposed RBF based trajectories must be assessed and continually modified in order to find the ideal trajectory for planned deblurring. To do this, an optimization problem can be created using weights of the RBF as an input, and the assigned score as output to minimize. The RBF offsets are not modified, and instead are equally spaced across the timescale. This is because with a sufficiently large number of functions and sufficiently large shape factor, the RBF function summation can still achieve a wide range of trajectories. Constraints are also imposed on the position endpoints x_0 and x_f, velocity endpoints v_0 and v_f via a chosen velocity $\bar{v}$, and acceleration $a(t)$ through a maximum allowable acceleration a_{allow} determined by the performance limit of a mobile camera platform actuator. The optimization problem formation is shown below:

$$\underset{\boldsymbol{w}\in R^n}{\text{minimize}}\ R_{OCR}(\boldsymbol{w}) \tag{3.9.1}$$

$$\text{subject to } x_0 = 0 \tag{3.9.2}$$

$$x_f = d_{\text{exp}} \tag{3.9.3}$$

$$v_i = v_f = \bar{v} \tag{3.9.4}$$

$$\max(a(t)) \leq a_{\text{allow}}, \forall t \in [0, T_e] \tag{3.9.5}$$

where $\boldsymbol{w}$ is the RBF weights vector:

$$\boldsymbol{w} = (w_1, w_2, ...w_n)^T. \tag{3.9.6}$$

The position vector is calculated as the numerical integral of the velocity vector $x(t) = \int_0^{\Delta T} v(t)dt$ while the acceleration is the numerical derivative $a(t) = \frac{d}{dt}v(t)$. Maximum acceleration is defined as the largest acceleration value over $[0, T_e]$. a_{allow} is chosen based upon the limits of the physical system for which the trajectory optimization is to be applied and imposes a non-linear constraint in this problem.

The velocity profile is the function generated by the RBF network so that the position endpoints can easily be constrained. The total distance traveled for any RBF is the area under the given function (in this case, the Gaussian) multiplied by the sum of all weights. From Equation 3.8:

$$d_{\text{exp}} = \left(\int_{T_e} e^{-(\epsilon r)^2} dr \sum_{k=1}^{n} w_k \right) + \bar{v}\Delta T. \tag{3.10}$$

The $\bar{v}\Delta T$ term accounts for the distance traveled using a linear trajectory with the same endpoint constraints. This equation can be used as a linear constraint to ensure the intended distance traveled is always met. The cost function $R_{OCR}(\boldsymbol{w})$ is the percent of incorrect characters in the text recognized string compared to the correct string measured by the Levenshtein distance algorithm.

3.5.1 Image Evaluation

Determination of an image evaluation method is a common issue for problems involving image restoration [40]. For this experiment, the returned character recognition text is compared to the ground truth text to form the rate of incorrect text recognition as determined by the Levenshtein distance algorithm [41]. This algorithm determines the number of insertions, deletions, or character changes needed to be applied to a string in order to make it match the goal. The number of changes is divided by the ground truth string length to create an incorrect text recognition rate:

$$R_{\text{OCR}}(\boldsymbol{w}) = \frac{n_{\text{L}}}{l_{\text{Text}}}, \tag{3.11}$$

where R_{OCR} is the rate, n_{L} is the number of string changes according to the Levenshtein distance algorithm, and l_{Text} is the length of the ground truth text of the image. One issue with this evaluation metric is the possibility of a significant difference in returned and actual string length, which may cause R_{OCR} to be significantly greater than 1. For this reason, a response of no text being recognized in the image is given a score of 3 to encourage the simulation to move towards any text recognition even if it contains many extra or missing characters. Any R_{OCR} over 3 are also reduced to 3 as well. Finally, any score values under 3 are scaled based on the mean R_{OCR} value of the ground truth image as a correction for some images being inherently more difficult to extract text from. The values of the mean recognition rate for these images can be found in Table A1.

3.5.2 Optical Character Recognition

Optical Character Recognition (OCR) is the process of converting text within images into machine-encoded text. A well known and widely used OCR software is Google's open source Tesseract [22], which was used within MATLAB's optimization process. Although many OCR algorithms exist, this experiment required the use of one that could be auto-

mated to run continuously without user input, making Tesseract an obvious choice. This engine performs well with its primary use for document text, but can struggle with natural scene text. A number of methods have previously been implemented in order to tailor OCR engines to natural scenes [24, 25]. Here, any input image is simply converted to gray scale and given to the OCR. This process is done automatically for every image during the optimization cycle.

3.6 Spectral Analysis

Spectral analysis is the process of deconstructing a signal in the time domain into components in the frequency domain. For the purposes of this study, a two-dimensional Discrete Fourier Transform is used to view and analyze frequency information within the both static and deblurred images. Specifically, edges, such as those formed by text, appear as high frequency values in the direction perpendicular to the edge in the two-dimensional power spectrum [40]. In the case of this one-dimensional, horizontal motion, the appearance and retention of high frequency values in the center horizontal region is critical to OCR's performance [36]. A methodology proposed here is to compute the difference between the power spectrum of a ground truth and a deblurred image power spectrum as shown in Figure 3.5. All images evaluated are proportionally resized to a 600 by 600 pixel image in order to maintain consistency across all samples. The resulting power spectrum difference describes the frequency information lost through the blurred capture and deblurring process. In order to convert this into a quantitative metric, the difference across all indices can be summed to find a total intensity difference. However, due to the limitation of the motion applied being one-dimensional in the horizontal direction, only the middle third of the image is considered as the horizontal edges in the image are not subject to blur. This yields a singular measurement of vertical edge information lost in the image which can be directly compared with OCR recognition rates.

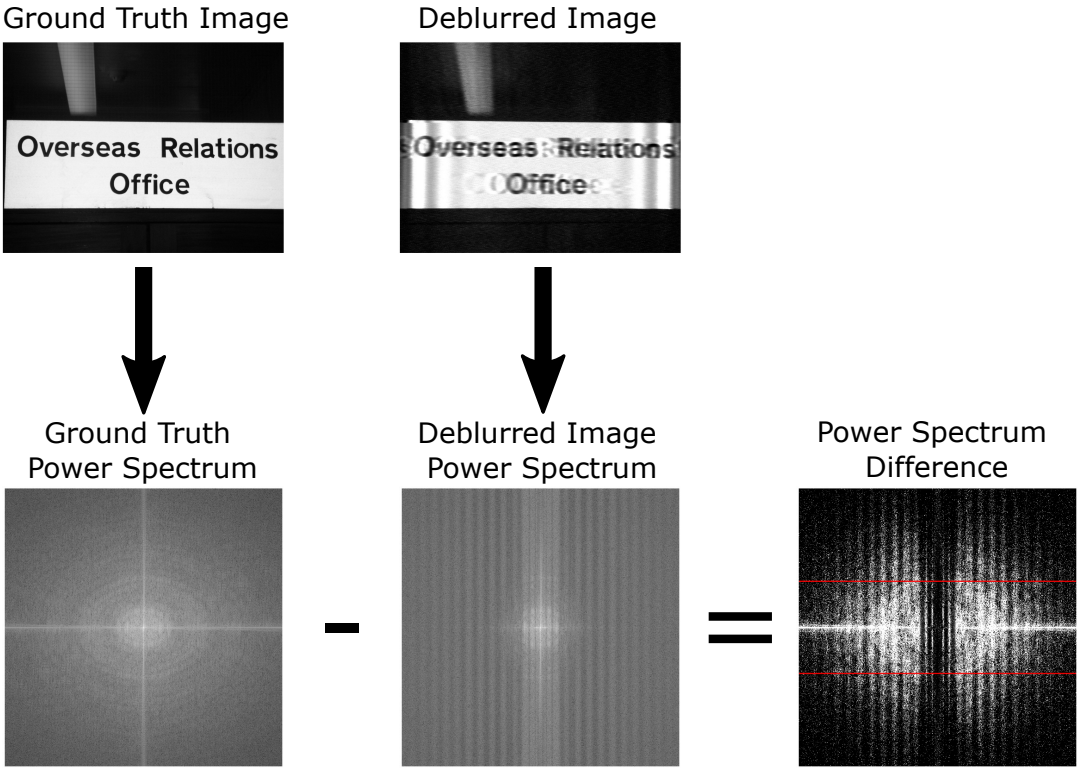

Figure 3.5: Power spectra of a static and deblurred image and the formation of the difference power spectrum with separated central horizontal third

CHAPTER 4
SIMULATION

4.1 Simulation Setup

Determining an ideal trajectory for character recognition requires an automated optimization process that can quickly evaluate many sets of RBF weights. A simulation of the process of capturing an image, deblurring it based on a known trajectory, and evaluating it was created in MATLAB. For any single iteration, a ground truth image, in this case, an image from the image library, is supplied and subjected to horizontal linear motion blur determined by the motion of an RBF trajectory with arbitrary weights. Noise is added to the blurred image, and the image is then deblurred with Richardson-Lucy deconvolution using the same blur kernel as the original blurring. OCR is then completed on the image, and the extracted text is compared to a correct text string using Levenshtein distance. The score is calculated using Equation 3.11 and scaled based on the static image's mean OCR score. The entire scoring process is completed for each image in the set using the same trajectory, and each individual image score is averaged for the set to form the set score. The set score is returned to MATLAB's genetic algorithm to determine the next suitable trajectory parameters to test. This process is repeated until a trajectory is found that satisfies the constraints in Equation 3.9 as well as the optimization parameters. Figure 4.1 shows this optimization algorithm in flowchart form.

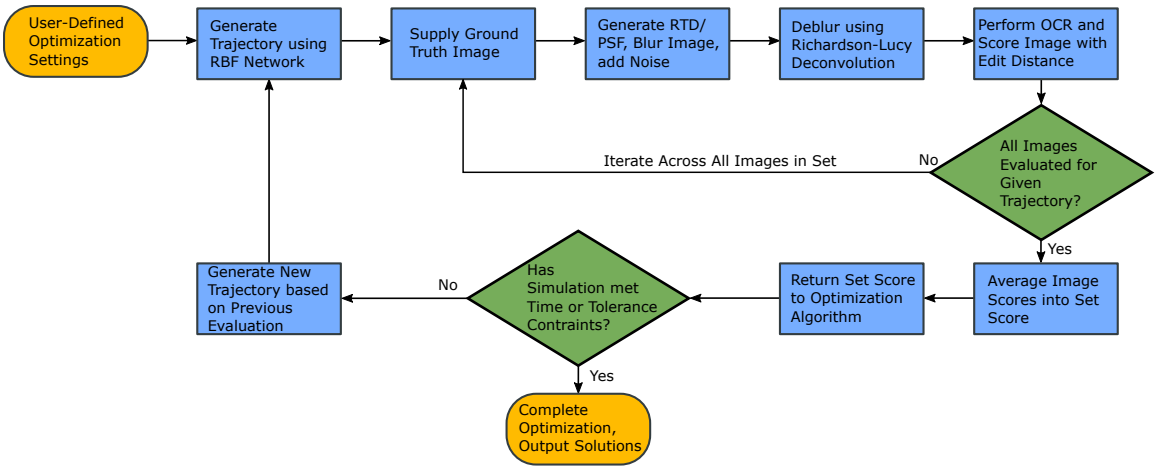

Figure 4.1: Flowchart of the simulation's optimization algorithm

One issue with the genetic algorithms is that it takes a very long time to converge while not necessarily providing significant improvement over time. Here, the goal of the genetic algorithm is to function as a random walk, so for this reason, a run-time limit of 1 hour was applied to each optimization. Based on the results obtained from the simulation being similar in shape, this did not affect the overall quality of the simulation's results.

Because individual natural images have only a small portion of text, a set of images is used for the optimization. For this experiment, a combination of the training and test set of images from the ICDAR 2015 OCR competition as well as the Street View Text Dataset was used [42, 43]. This comprises a library of 812 images, however, this was reduced down to 27 by filtering for images that scored an average of no greater than 0.5 in R_{OCR} value when evaluated as static images. This was done to prevent an overly high rate of incorrect recognition from images that would be particularly difficult for the OCR engine. The final set contains 788 total characters. The image library can found in Figure A1.

Three different sets of simulations were completed. The first is the main optimization comprising a single large search from which the best five unique trajectories were selected. Linear and inverse error trajectories were also tested as a comparison method, with the inverse error function defined as the inverse of the Gaussian error function. For each trajectory, five trials of the image set were run in order to mitigate variability introduced by

the OCR engine. The parameters for this simulation are shown in Table 4.1.

Table 4.1: Main search simulation parameters

Simulation Parameter	Value
Number of Images	27
Maximum Run Time	1 hour
Simulated Image Width	0.5 m
Number of RBF Functions	13 (7 unique)
ϵ	4
$\bar{v}$	0.2 m/s
a_{max}	$5g$
ΔT	0.2 s (5 FPS)
d_{exp}	40 mm

The number of RBFs and ϵ value were chosen based upon what combination could reasonably be used to fully represent almost any trajectory between endpoints while reducing computation time. $\bar{v}$ is based upon the linear velocity needed to meet position endpoint constraints and a_{max} was chosen based upon what could be achieved with the hardware setup. Simulated image width, d_{exp}, and camera exposure ΔT were also based on experiment setup.

The second series of simulation runs attempts to draw a relationship between the acceleration limit of the optimization and the overall score. The optimization search was completed four times using the same parameters above except for the acceleration limit, which was changed to $1g$, $2g$, $3g$, and $4g$. For every search, five trials were run, just as above. The best performing $5g$ test is used for comparison.

The last set of tests again used the same simulation parameters as the first test except for a change in camera exposure time and corresponding endpoint velocity. Four searches were completed with exposure times of $1/15$, $1/30$, $1/45$, and $1/60$ seconds with endpoint

velocities necessary to create a linear trajectory. Once again, the best non-linear trajectory from each search was selected to be run for five trials.

4.2 Simulation Results

A representation of the first optimization is shown in Figure 4.2. The plots show the scores for every image as well as the output score used to generate new outputs by the genetic algorithm. The circled low scores are the trajectories pulled from the optimization for further testing (some points are skipped due to being identical to other already selected points). The bottom plots show the range of trajectories attempted by the optimization. This representation is similar to the optimization run for the rest of the attempted simulations.

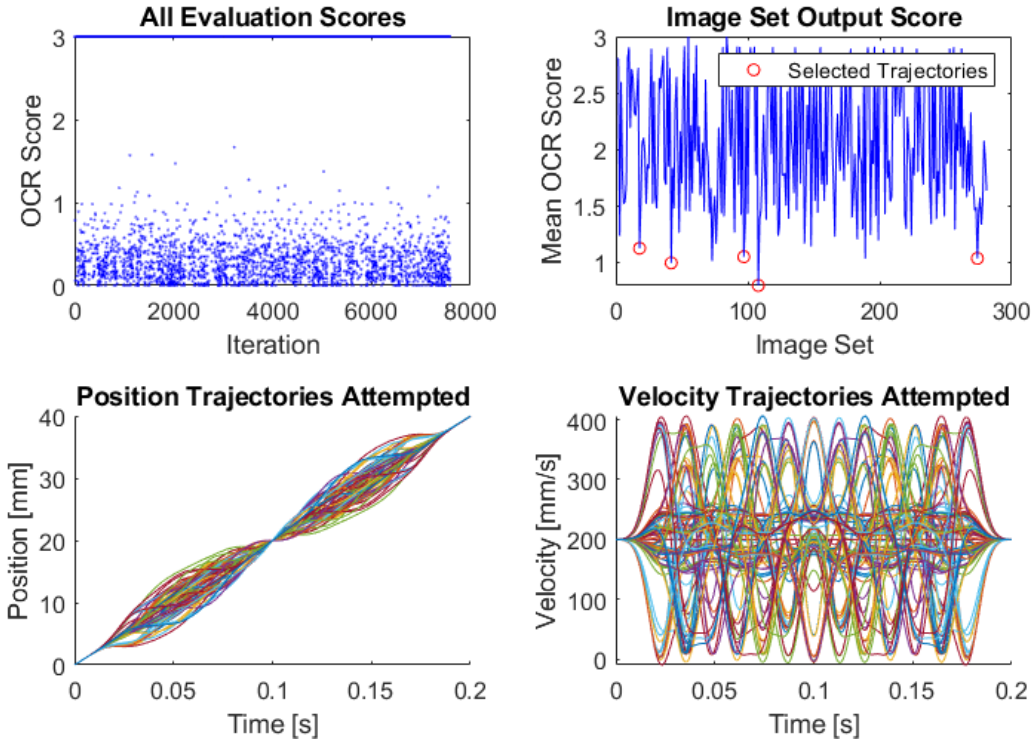

Figure 4.2: Output plots of the optimization search including individual and set scores as well as trajectories attempted

Sample results for the first image in the image set are shown in Figures 4.3 to 4.9 for the linear, inverse error, and RBF trajectories 1 through 5 as well as their position, velocity, and residence time distribution profiles.

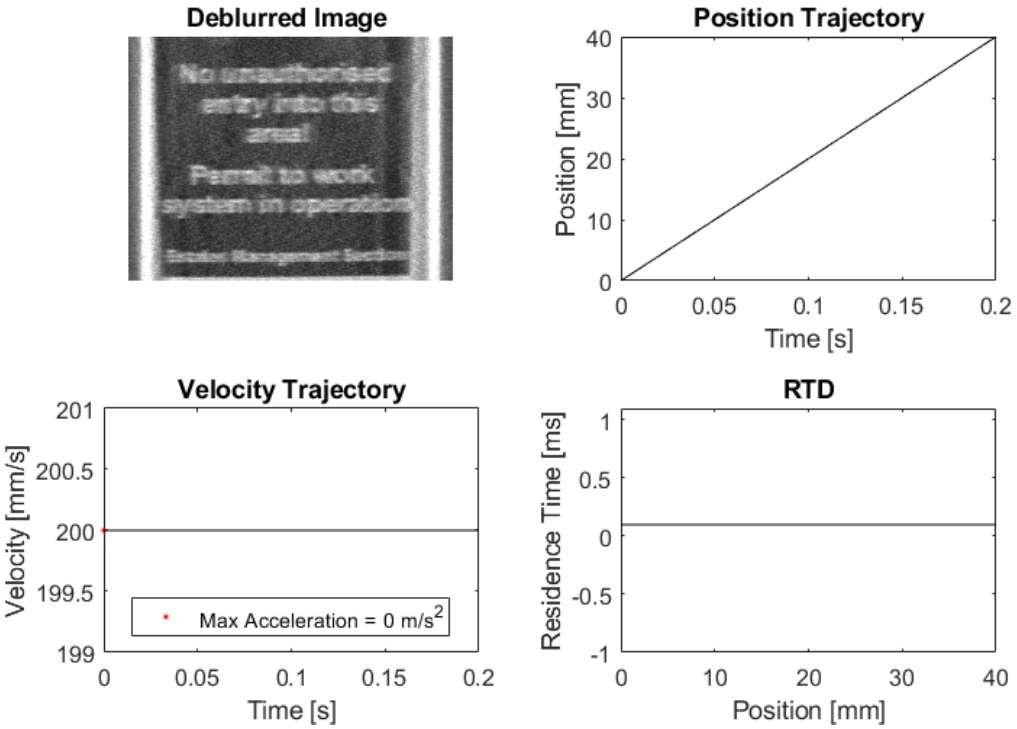

Figure 4.3: Sample of simulated deblurred image using linear trajectory with position, velocity, and residence time distribution plots

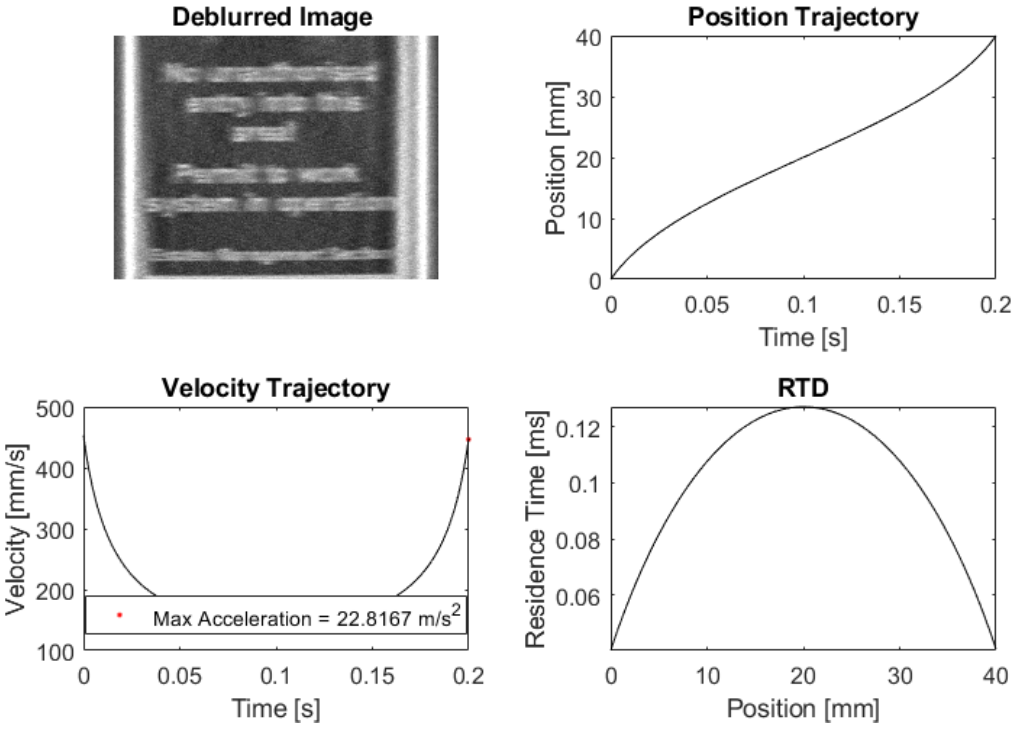

Figure 4.4: Sample of simulated deblurred image using inverse error trajectory with position, velocity, and residence time distribution plots

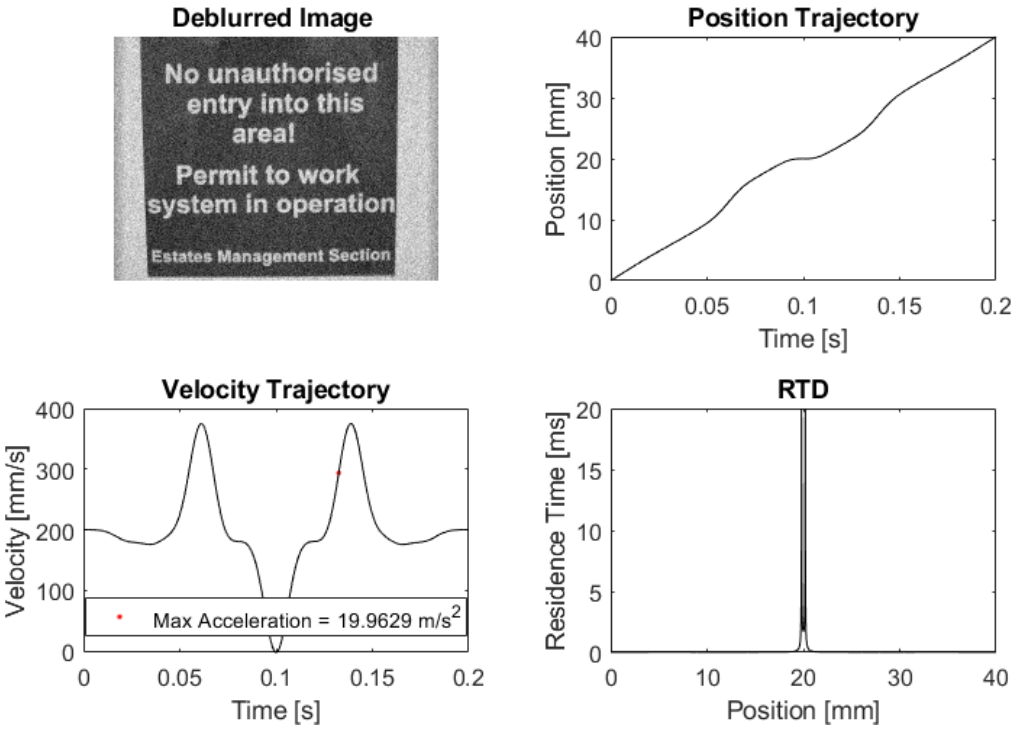

Figure 4.5: Sample of simulated deblurred image using RBF1 trajectory with position, velocity, and residence time distribution plots

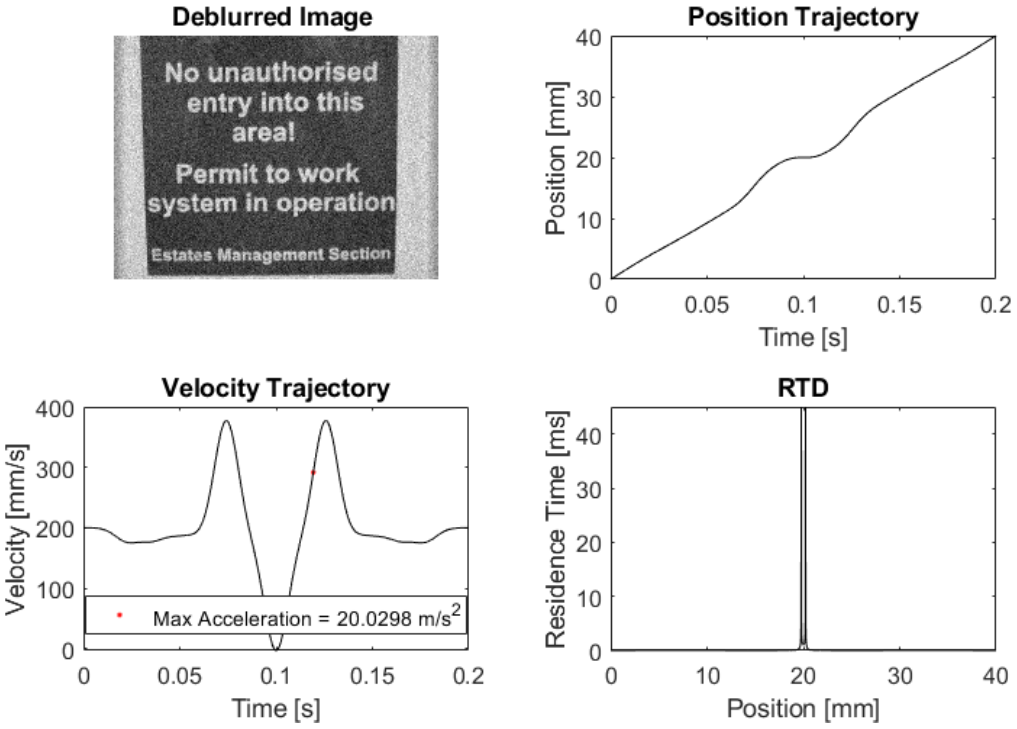

Figure 4.6: Sample of simulated deblurred image using RBF2 trajectory with position, velocity, and residence time distribution plots

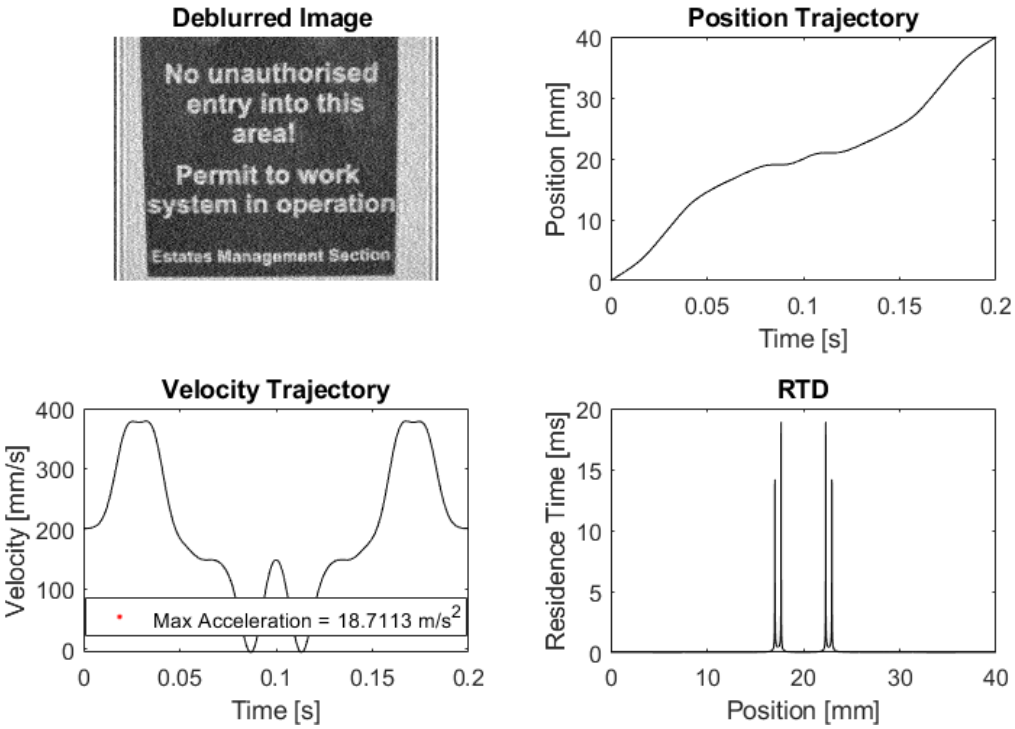

Figure 4.7: Sample of simulated deblurred image using RBF3 trajectory with position, velocity, and residence time distribution plots

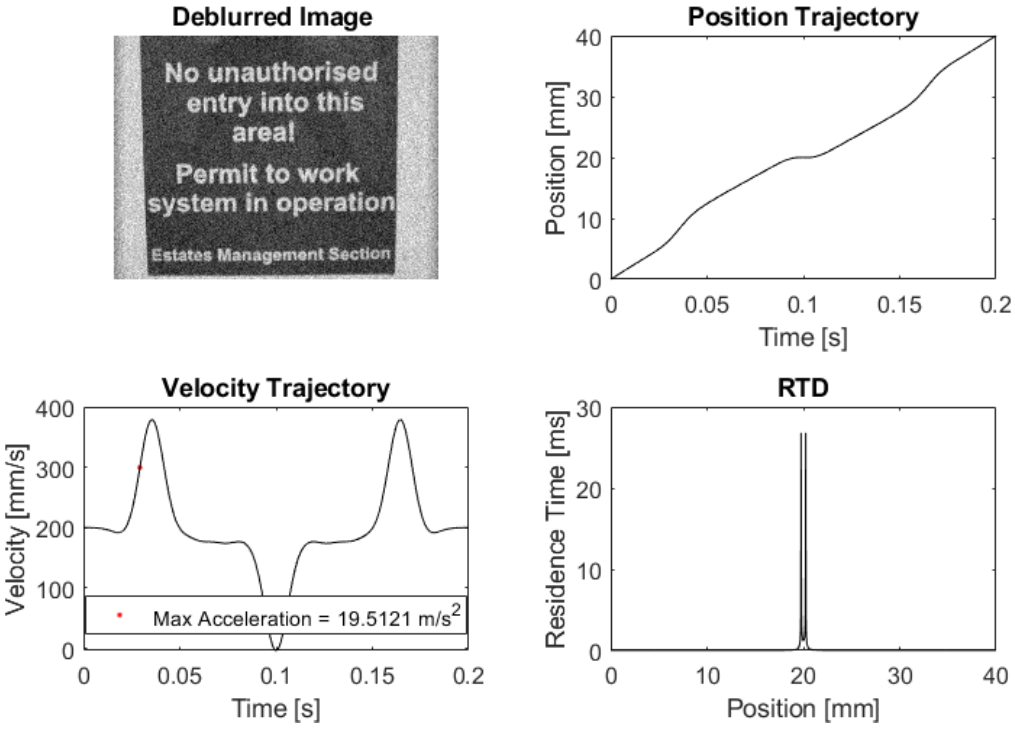

Figure 4.8: Sample of simulated deblurred image using RBF4 trajectory with position, velocity, and residence time distribution plots

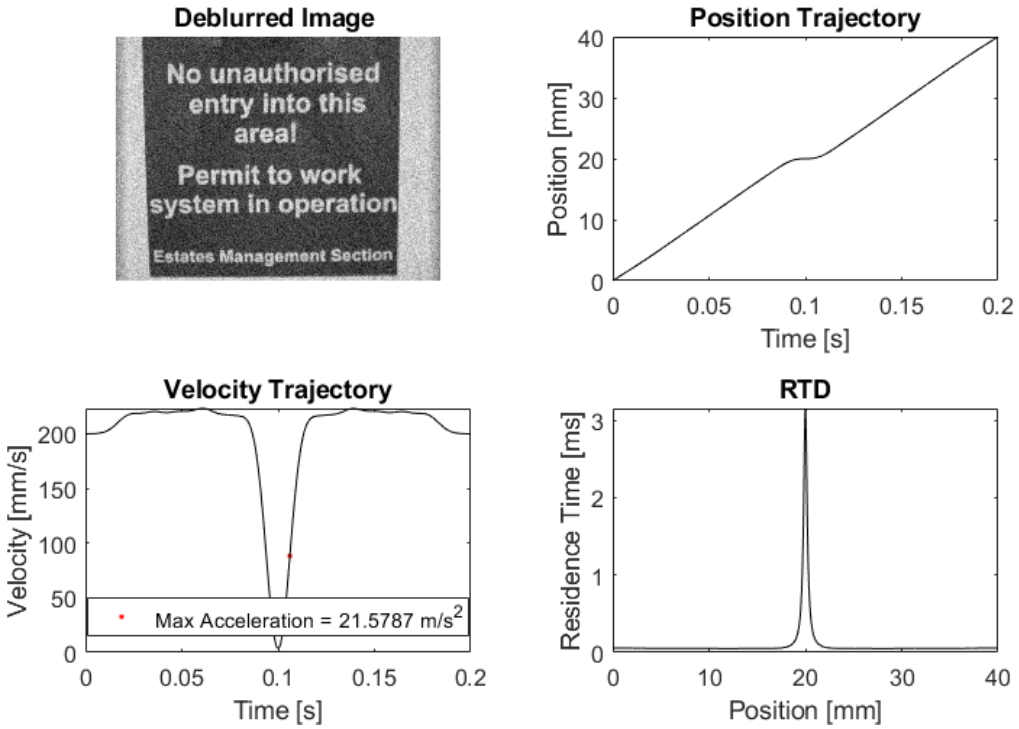

Figure 4.9: Sample of simulated deblurred image using RBF5 trajectory with position, velocity, and residence time distribution plots

The RBF trajectories shown above represent potential optimal trajectories given the study parameters and constraints used in the optimization. It is immediately noticeable that all of the deblurred images from the RBF trajectories are less blurred than those of the linear and inverse error trajectories, indicating the optimization has yielded improved results compared to the default linear trajectory. There is also a general trend of the RBF trajectories having either single or multi-peak RTDs around the center of the trajectory, which represents the camera stalling at the image center. This is confirmed in the velocity plots which show the camera rapidly moving to the image center, stopping or significantly slowing, and then rapidly moving to the end of the trajectory.

Table 4.2 shows the mean, standard deviation, and t-test values compared to the linear trajectory of all trajectories across all images and trials. All t-tests for this and other comparisons are left-tailed tests to compare if the suspected trajectory performs better than the baseline trajectory. The RBF trajectories significantly outperform the linear and inverse

error, up to 57.3% when comparing the best performing RBF4 to linear, all with $p \ll 0.05$ statistical significance. One other interesting trend is a relationship between the highest time achieved in the RTD and the overall performance. RBF 4 and 2 are the best scoring trajectories and have the highest RTD times with the remaining trajectories following in order of highest RTD time. Sample simulation score and OCR data for RBF4 can be found in Table A2.

Table 4.2: Main search simulation score means μ_{OCR}, standard deviations σ_{OCR}, and t-test scores t_{OCR} by trajectory over five trials

Trajectory	μ_{OCR}	σ_{OCR}	t_{OCR}
Linear	2.684	0.121	-
Inverse	2.872	0.082	0.974
RBF1	1.180*	0.232	1.241E-04
RBF2	1.196*	0.103	4.715E-05
RBF3	1.200*	0.183	3.510E-05
RBF4	1.146*	0.150	1.002E-06
RBF5	1.362*	0.103	5.299E-05

Note: $*p < 0.05$ compared to linear trajectory

In addition to the above calculations, all t-test p-values comparing each trajectory to each other are shown in Table 4.3. This is presented in order to compare the performance between RBF trajectories; the trajectory on the row is being tested for statistical significance against the trajectory on the column. Ultimately, no RBF trajectories show significantly different performance than others except for some statistical significance of RBF5 being slightly under performing compared to the other RBF trajectories.

Table 4.3: Main search simulation t-test p-values matrix by trajectory

Trajectory	Linear	Inverse	RBF1	RBF2	RBF3	RBF4	RBF5
Linear	-	0.026*	1	1	1	1	1
Inverse	0.974	-	1	1	1	1	1
RBF1	1.241E-04*	1.783E-05*	-	0.454	0.458	0.621	0.032*
RBF2	4.715E-05*	7.284E-06*	0.546	-	0.486	0.660	0.029*
RBF3	3.510E-05*	4.574E-05*	0.542	0.514	-	0.683	0.133
RBF4	1.002E-06*	6.726E-06*	0.379	0.340	0.317	-	0.035*
RBF5	5.299E-05*	8.215E-07*	0.968	0.971	0.867	0.965	-

Note: *$p < 0.05$

The output plots shown in Figures 4.10 to 4.13 show sample deblurred images for trajectories attempted with various acceleration limits.

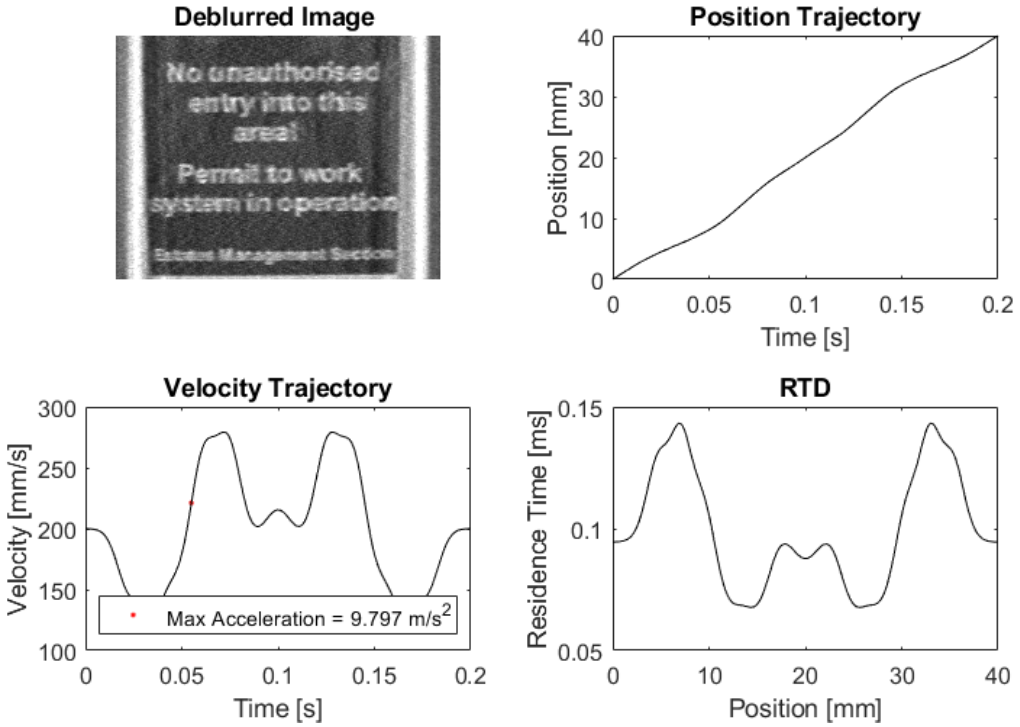

Figure 4.10: Sample of simulated deblurred image using an optimized trajectory for a $1g$ acceleration limit with position, velocity, and residence time distribution plots

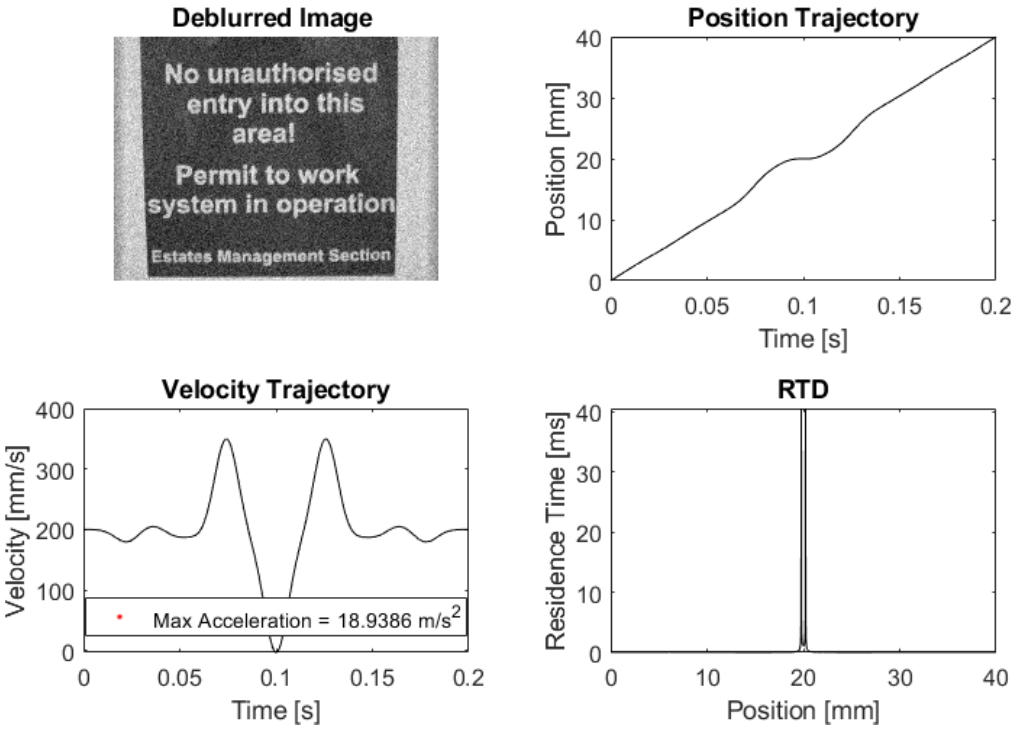

Figure 4.11: Sample of simulated deblurred image using an optimized trajectory for a $2g$ acceleration limit with position, velocity, and residence time distribution plots

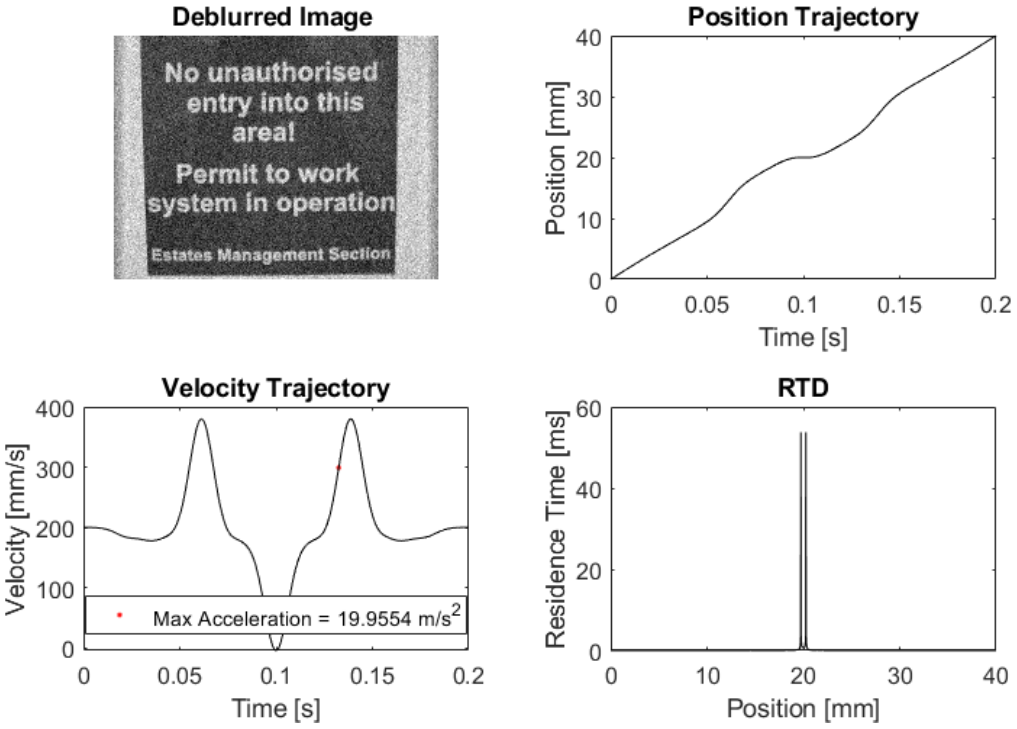

Figure 4.12: Sample of simulated deblurred image using an optimized trajectory for a $3g$ acceleration limit with position, velocity, and residence time distribution plots

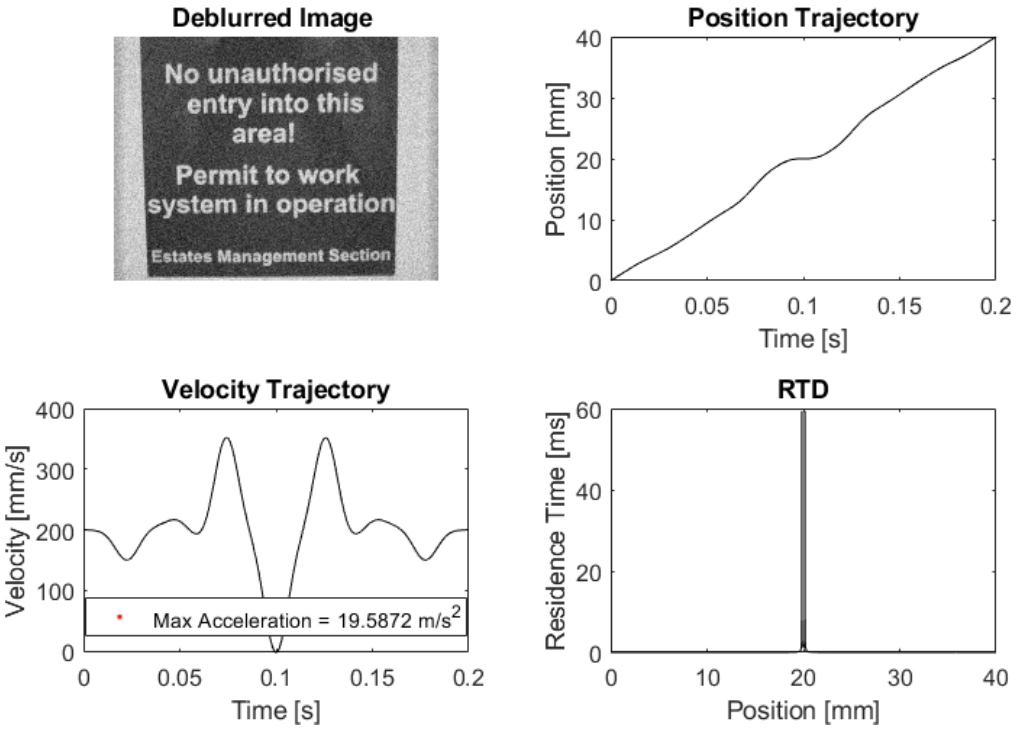

Figure 4.13: Sample of simulated deblurred image using an optimized trajectory for a $4g$ acceleration limit with position, velocity, and residence time distribution plots

The score results from these four trajectories are tabulated in Table 4.4 with the best performing RBF trajectory from the original $5g$ set, RBF 4, being used for comparison. Once again, all trajectories have $p \ll 0.05$ statistical significance compared to the linear trajectory. The results from this set of trajectories are very similar to the results of the first set in both shape and RTD values, except for the trajectory with a $1g$ acceleration limit. At this limit, the score significantly deteriorates as the velocity can no longer sustain high or low enough speeds to create peaks in the RTD. Instead, very short double peaks appear outside of the center. This creates strong ringing artifacts and worse deblurring compared to the other trajectories.

Table 4.4: Acceleration limit simulation score means μ_{OCR}, standard deviations σ_{OCR}, and t-test scores t_{OCR} by trajectory over five trials

Trajectory	μ_{OCR}	σ_{OCR}	t_{OCR}
$1g$ Limit	2.396*	0.131	1.251E-02
$2g$ Limit	1.188*	0.125	2.210E-05
$3g$ Limit	1.300*	0.133	2.083E-06
$4g$ Limit	1.067*	0.174	3.380E-06
$5g$ Limit	1.146*	0.150	1.002E-06

Note: *$p < 0.05$ compared to linear trajectory

The t-test p-value matrix for these tests is shown in Table 4.5. From the table, the lack of deblurring ability in the $1g$ case is confirmed as all trajectories perform better than it. The $4g$ and $5g$ case also show improvement over the $3g$ case, which is expected.

Table 4.5: Acceleration limit simulation t-test p-values matrix by trajectory

Trajectory	$1g$	$2g$	$3g$	$4g$	$5g$
$1g$	-	1	1	1	1
$2g$	3.452E-05*	-	0.155	0.881	0.661
$3g$	2.602E-04*	0.845	-	0.969	0.980
$4g$	1.451E-05*	0.119	0.031*	-	0.113
$5g$	7.513E-05*	0.339	0.020*	0.887	-

Note: *$p < 0.05$

The final set of optimized trajectories for various exposure times is shown below in Figures 4.14 to 4.17.

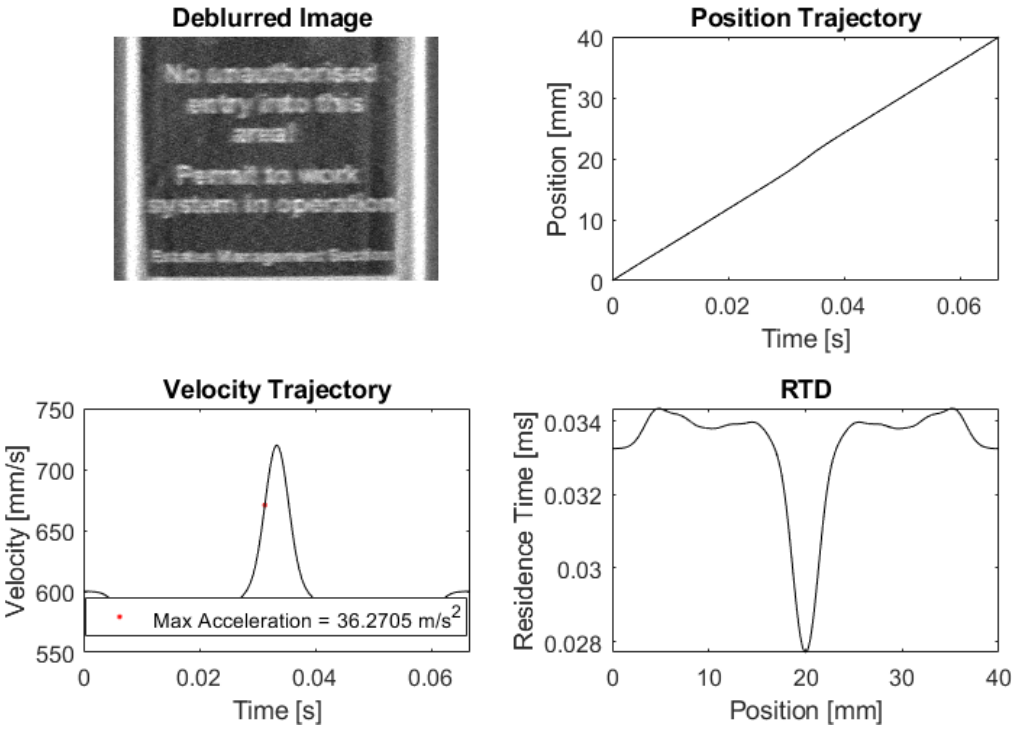

Figure 4.14: Sample of simulated deblurred image using an optimized trajectory for a a 1/15 second exposure time with position, velocity, and residence time distribution plots

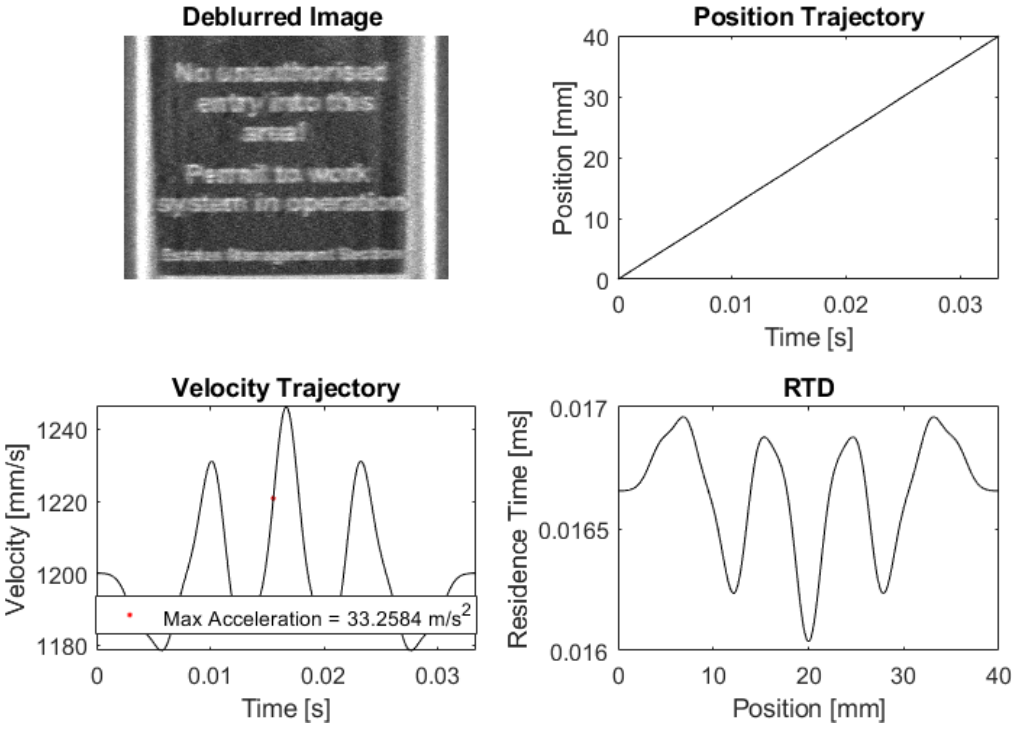

Figure 4.15: Sample of simulated deblurred image using an optimized trajectory for a a 1/30 second exposure time with position, velocity, and residence time distribution plots

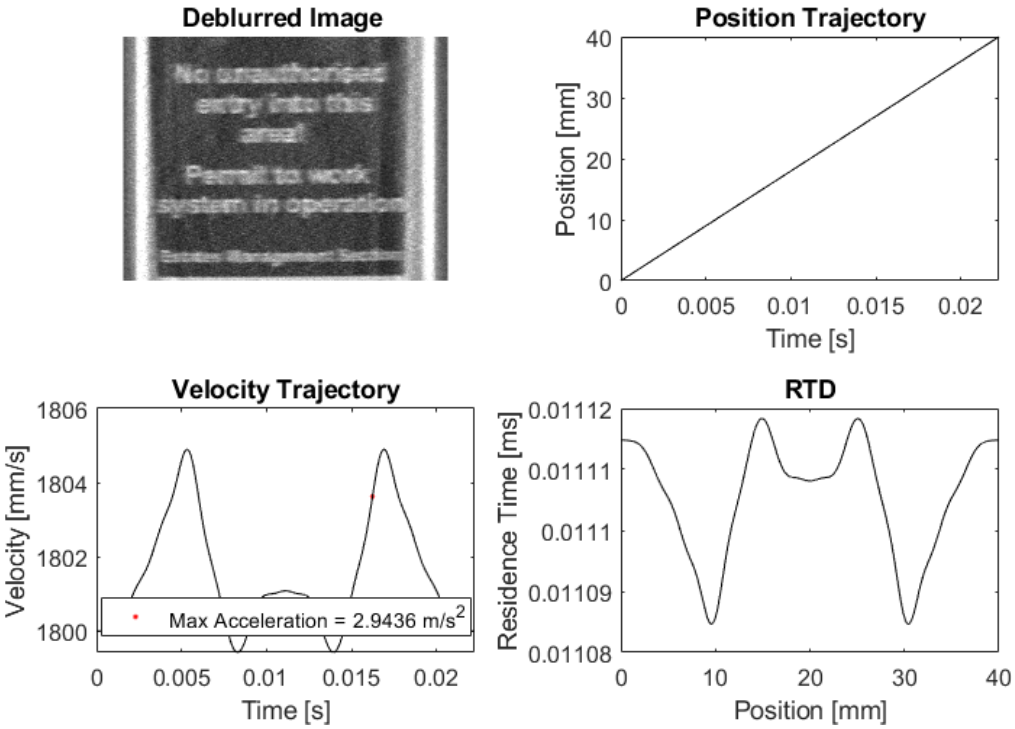

Figure 4.16: Sample of simulated deblurred image using an optimized trajectory for a a 1/45 second exposure time with position, velocity, and residence time distribution plots

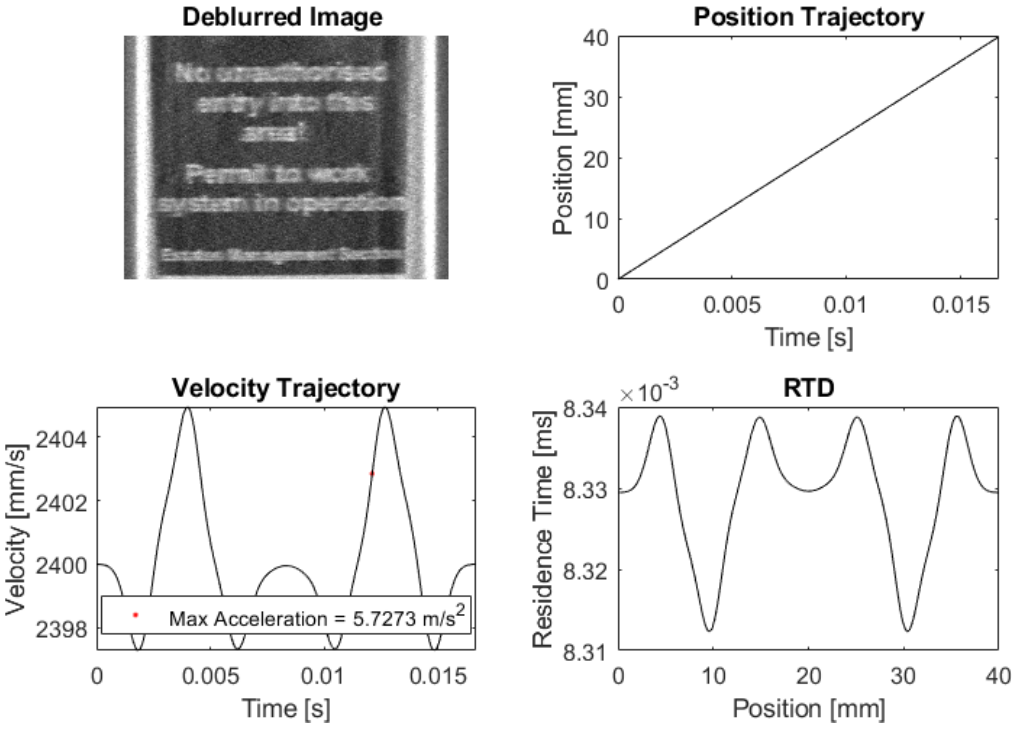

Figure 4.17: Sample of simulated deblurred image using an optimized trajectory for a a 1/60 second exposure time with position, velocity, and residence time distribution plots

These trajectories cause the images to remain heavily blurred because the trajectories must meet increasingly more difficult endpoint constraints. Summary values in Table 4.6 confirm that these trajectories perform similarly to a linear trajectory as they are essentially linear. However, it does appear that these trajectories did attempt to create peaks in the RTD with the difference between the maximum and minimum value decreasing with a decrease in exposure time. The p-value matrix in Table 4.7 shows no major difference in performance between these trajectories.

Table 4.6: Exposure time simulation score means μ_{OCR}, standard deviations σ_{OCR}, and t-test scores t_{OCR} by trajectory over five trials

Trajectory	μ_{OCR}	σ_{OCR}	t_{OCR}
15 FPS	2.727	0.077	0.812
30 FPS	2.766	0.044	0.898
45 FPS	2.728	0.153	0.715
60 FPS	2.701	0.070	0.585

Note: $*p < 0.05$ compared to linear trajectory

Table 4.7: Exposure time simulation t-test p-values matrix by trajectory

Trajectory	15FPS	30FPS	45FPS	60FPS
15FPS	-	0.086	0.495	0.679
30FPS	0.914	-	0.688	0.900
45FPS	0.505	0.312	-	0.641
60FPS	0.321	0.100	0.359	-

Note: $*p < 0.05$

4.3 Spectral Analysis

Spectral analysis was performed on all of the simulated trajectories. First, Figure 4.18 shows the power spectrum for the static image that is used in the following difference spectrum for each trajectory. The power spectrum of the deblurred image and its difference from the base power spectra for the linear, inverse error, and five base RBF trajectories are shown in Figures Figures 4.19 to 4.25.

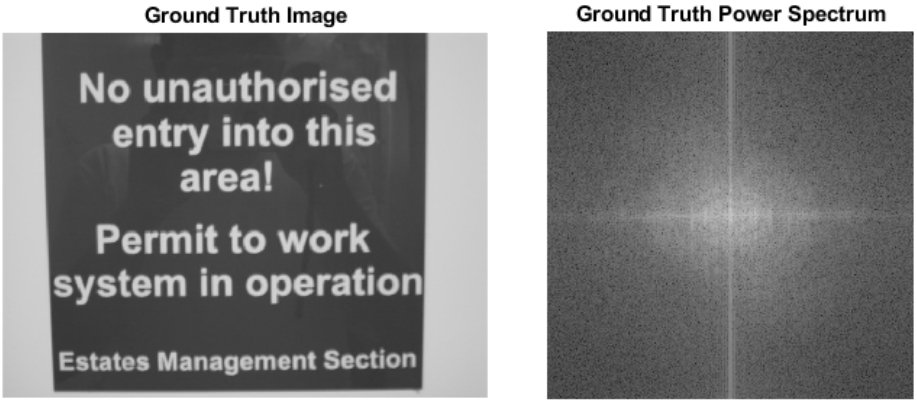

Figure 4.18: Static image 1 and its power spectra

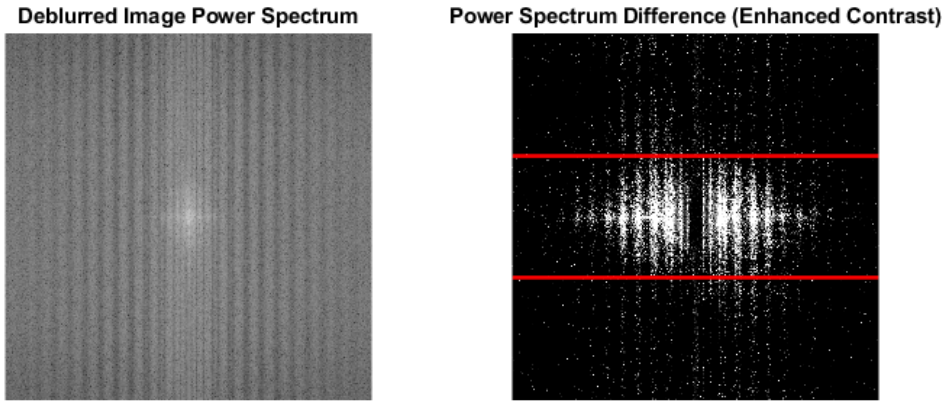

Figure 4.19: Deblurred image power spectra from simulated linear trajectory and power spectra difference from static image power spectra

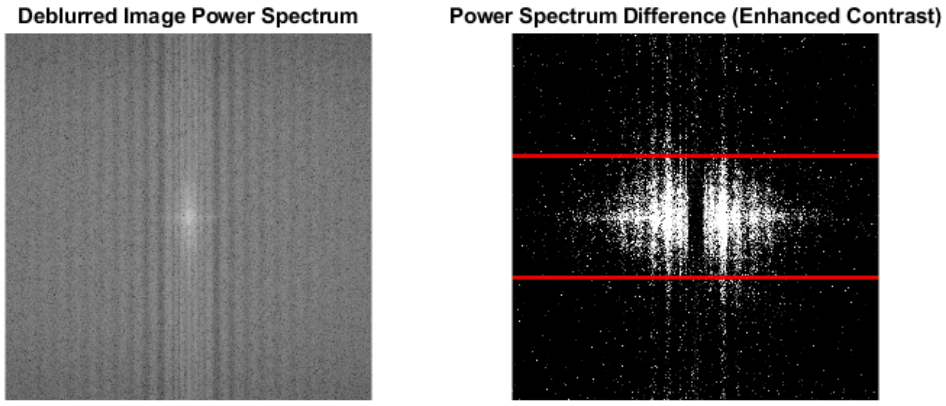

Figure 4.20: Deblurred image power spectra from simulated inverse error trajectory and power spectra difference from static image power spectra

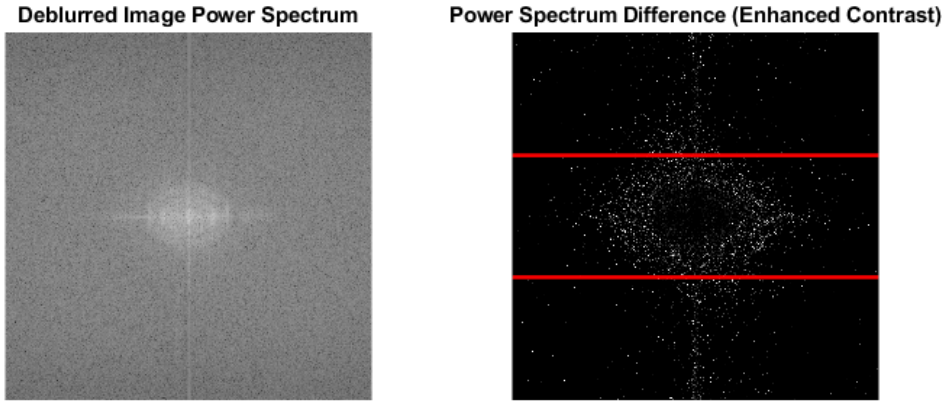

Figure 4.21: Deblurred image power spectra from simulated RBF1 and power spectra difference from static image power spectra

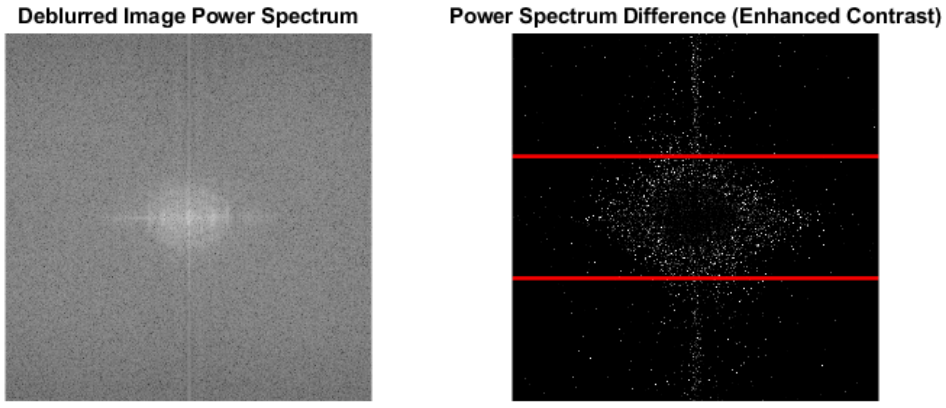

Figure 4.22: Deblurred image power spectra from simulated RBF2 and power spectra difference from static image power spectra

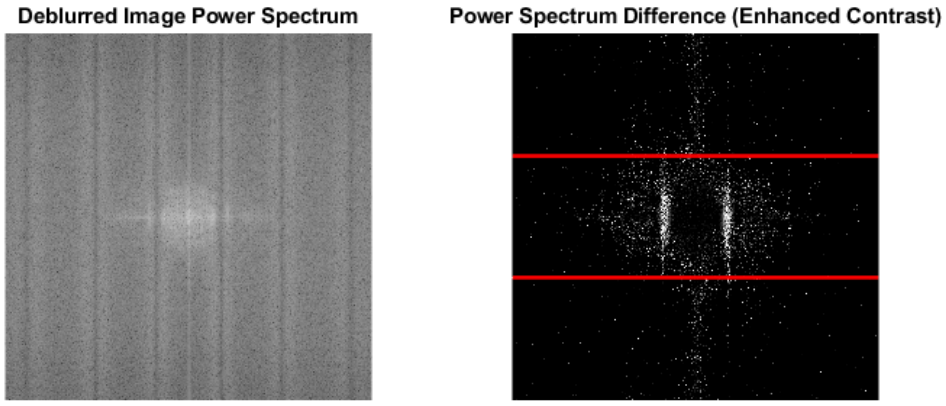

Figure 4.23: Deblurred image power spectra from simulated RBF3 and power spectra difference from static image power spectra

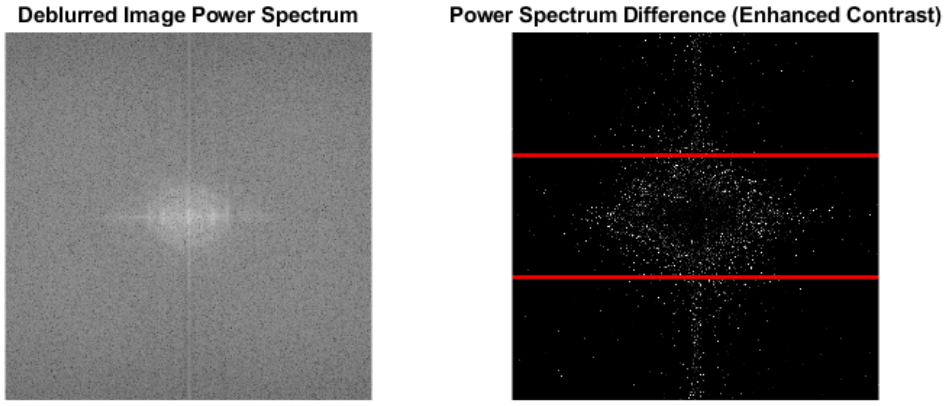

Figure 4.24: Deblurred image power spectra from simulated RBF4 and power spectra difference from static image power spectra

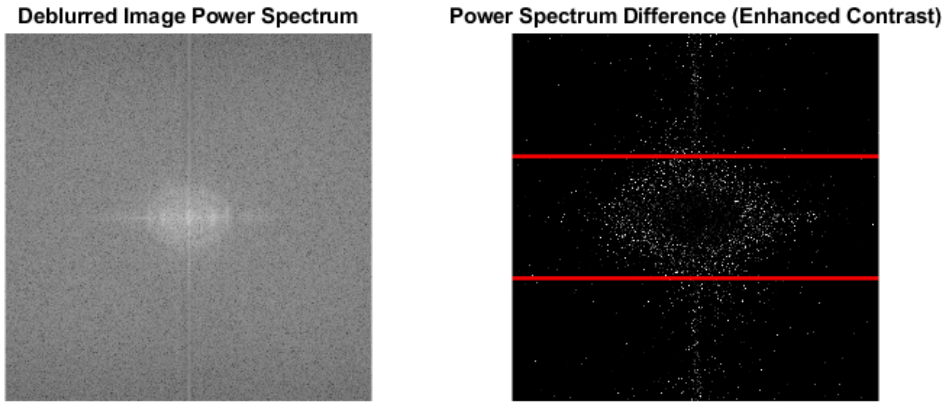

Figure 4.25: Deblurred image power spectra from simulated RBF5 and power spectra difference from static image power spectra

On the difference power spectrum, lines are imposed to show the central third of the image that most represents the relevant horizontal frequencies. The static image power spectrum shows high frequency components in both the purely vertical and horizontal directions representing the edges of the box in the image as well as the edges of the letters in these directions, especially for characters such as “N”, “d”, and “P”. There is also a large amount of lower frequency information in all directions representing the curved and slanted edges of the other letters. For the linear and inverse trajectory plots, there is a significant removal of these regions as this information has been distorted by the blur. The difference plots show loss of both the high frequency horizontal components as well as lower frequency near-horizontal regions. This loss is much less prevalent in all of the RBF trajectory power spectra. The sum of the intensity differences for the central horizontal third of each image is calculated in Table 4.8. This shows a very close correlation between the maintenance of these image frequencies and the text recognition rate undergoing one-dimensional blur.

Table 4.8: Main search simulated power spectra difference means μ_{PS}, standard deviations σ_{PS}, and t-test scores t_{PS} by trajectory over five trials

Trajectory	μ_{PS}	σ_{PS}	t_{PS}
Linear	7065.0	2.803	-
Inverse	7248.8	2.267	1
RBF1	5994.4*	1.176	3.645E-12
RBF2	6448.8*	3.048	8.861E-10
RBF3	6475.9*	0.439	5.517E-11
RBF4	6615.9*	1.411	3.701E-10
RBF5	6513.1*	2.756	1.756E-10

Note: $*p < 0.05$ compared to linear trajectory

The power spectra t-test p-value matrix is calculated in Table 4.9. The results here show significance in many tests, however, this is likely due to the low standard deviation of the

score and is not necessarily indicative of specific performance comparisons.

Table 4.9: Main search simulation power spectra score t-test p-values matrix by trajectory

Trajectory	Linear	Inverse	RBF1	RBF2	RBF3	RBF4	RBF5
Linear	-	2.520E-09*	1	1	1	1	1
Inverse	1	-	1	1	1	1	1
RBF1	3.645E-12*	1.356E-12*	-	3.488E-10*	1.127E-12*	1.159E-11*	3.067E-10*
RBF2	8.861E-10*	1.636E-10*	1	-	1.939E-05*	1.553E-08*	5.235E-06*
RBF3	5.517E-11*	1.028E-11*	1	1	-	2.546E-09*	5.226E-06*
RBF4	3.701E-10*	9.731E-12*	1	1	1	-	1
RBF5	1.756E-10*	3.426E-11*	1	1	1	1.065E-07*	-

Note: $*p < 0.05$

For the acceleration limit tests, similar results are present with the spectra difference sum being very closely related to the OCR score. Once again, the low acceleration limit test performed much more poorly than the other four trajectories. Figures 4.26 to 4.29 shows these results and Table 4.10 presents the statistical information relating to these values. In the p-value matrix in Table 4.11, all other trajectories showed statistical significance against the $1g$ case while no major correlation is seen among the other cases.

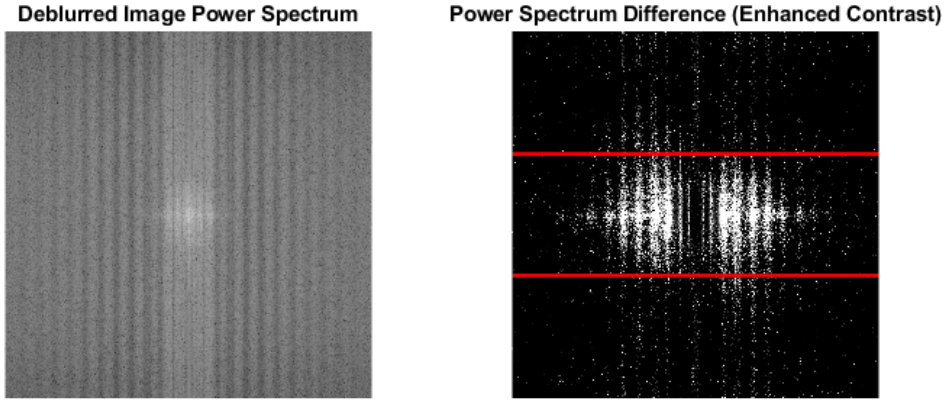

Figure 4.26: Deblurred image power spectra from simulated $1g$ acceleration limit trajectory and power spectra difference from static image power spectra

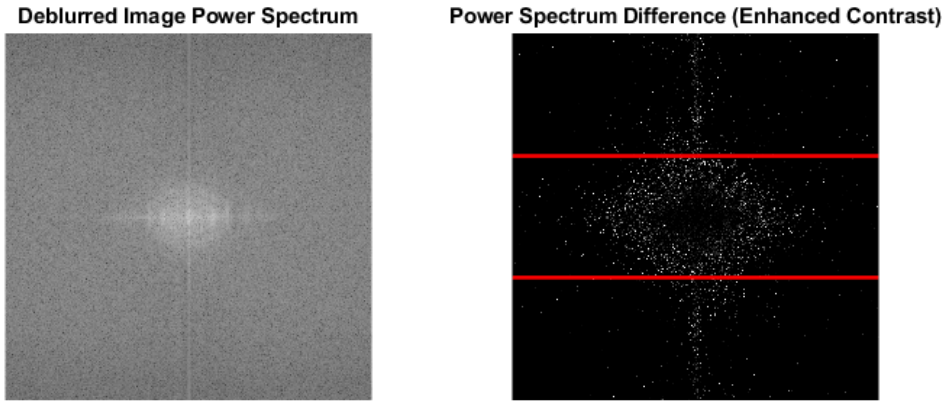

Figure 4.27: Deblurred image power spectra from simulated $2g$ acceleration limit trajectory and power spectra difference from static image power spectra

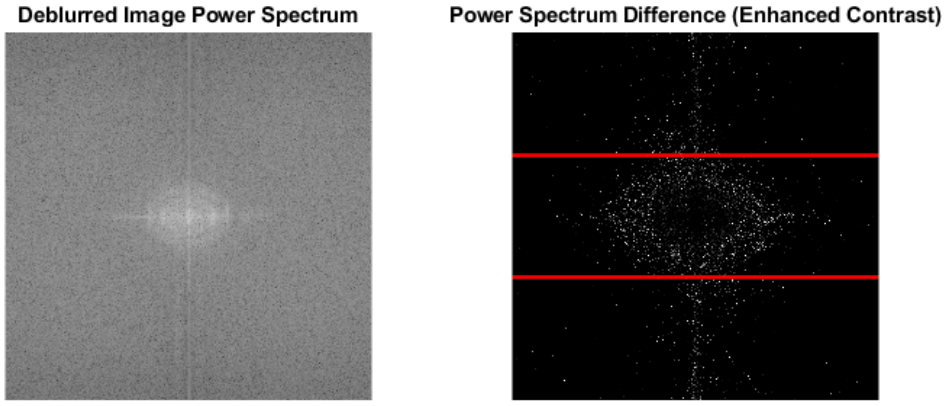

Figure 4.28: Deblurred image power spectra from simulated $3g$ acceleration limit trajectory and power spectra difference from static image power spectra

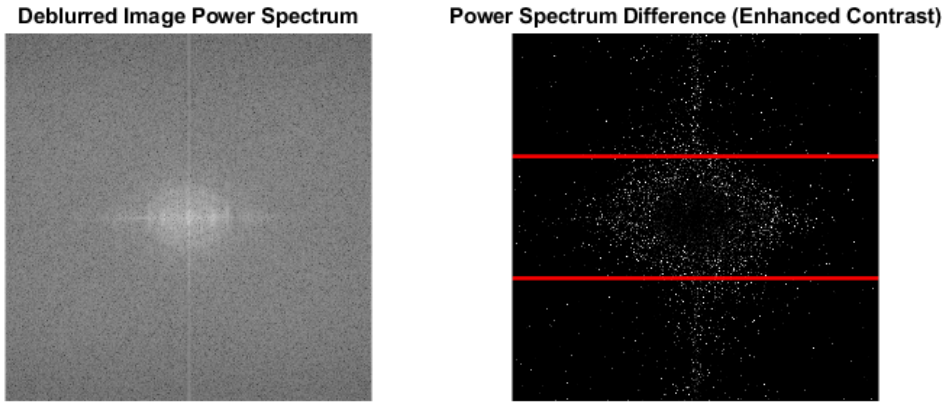

Figure 4.29: Deblurred image power spectra from simulated $4g$ acceleration limit trajectory and power spectra difference from static image power spectra

Table 4.10: Acceleration limit simulated power spectra difference means μ_{PS}, standard deviations σ_{PS}, and t-test scores t_{PS} by trajectory over five trials

Trajectory	μ_{PS}	σ_{PS}	t_{PS}
$1g$ Limit	7058.1*	2.684	3.166E-03
$2g$ Limit	6438.1*	1.242	8.014E-11
$3g$ Limit	6956.4*	2.134	9.979E-09
$4g$ Limit	5480.7*	2.814	2.641E-12
$5g$ Limit	6615.9*	1.411	3.701E-10

Note: *$p < 0.05$ compared to linear trajectory

Table 4.11: Acceleration limit simulation power spectra score t-test p-values matrix by trajectory

Trajectory	$1g$	$2g$	$3g$	$4g$	$5g$
$1g$	-	1	1	1	1
$2g$	4.012E-11*	-	9.841E-11*	1	1.212E-09*
$3g$	1.316E-08*	1	-	1	1
$4g$	1.338E-11*	1.878E-11*	6.726E-12*	-	3.606E-12*
$5g$	3.723E-10*	1	8.385E-10*	1	-

Note: *$p < 0.05$

For the final set of exposure time tests, the power spectra plots very closely resemble that of the linear trajectory with similar losses of horizontal information and introduction of ringing artifacts as shown in Figures 4.30 to 4.33. Their power spectra difference scores are accordingly very similar to the linear trajectory due to the new power spectra being nearly indistinguishable from the linear power spectra, equating to a very significant loss of edges perpendicular to the horizontal direction. These scores are calculated in Table 4.12. p-value comparisons in Table 4.13 show some statistical significance but no major correlation

between the trajectories.

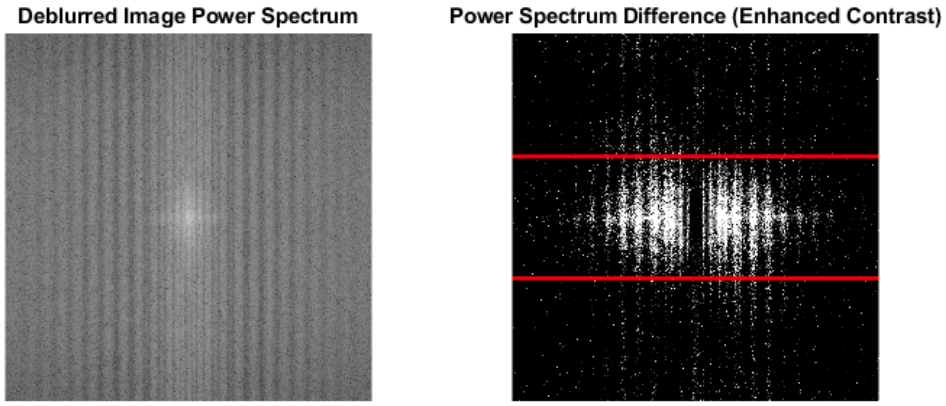

Figure 4.30: Deblurred image power spectra from simulated 1/15 second exposure time trajectory and power spectra difference from static image power spectra

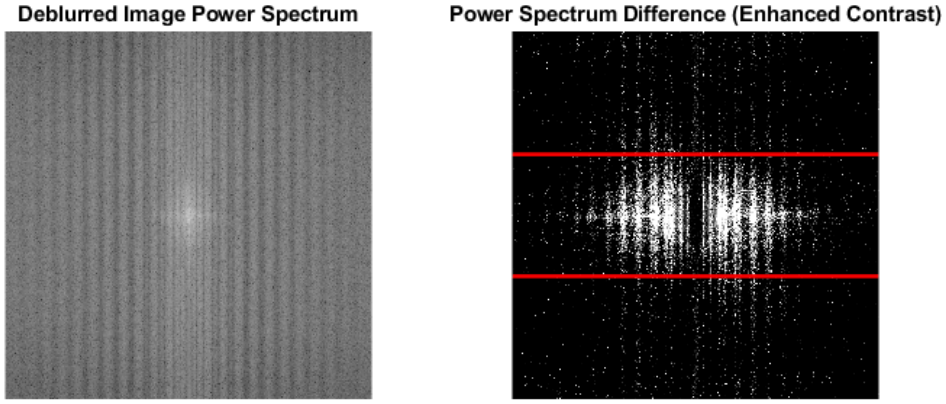

Figure 4.31: Deblurred image power spectra from simulated 1/30 second exposure time trajectory and power spectra difference from static image power spectra

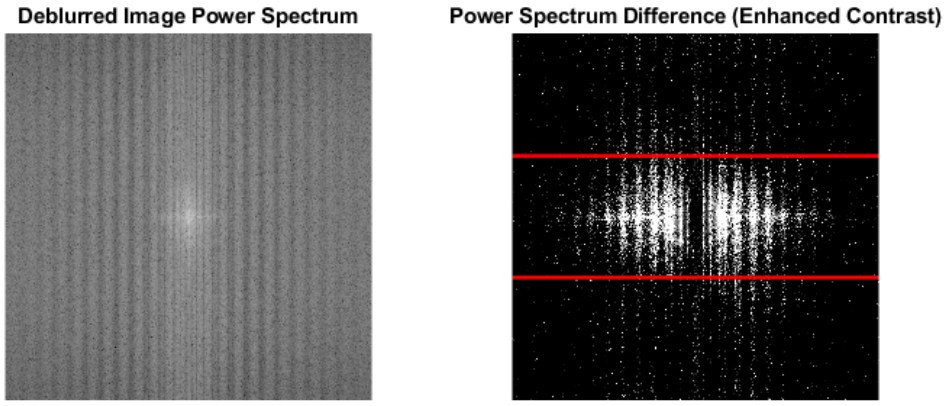

Figure 4.32: Deblurred image power spectra from simulated 1/45 second exposure time trajectory and power spectra difference from static image power spectra

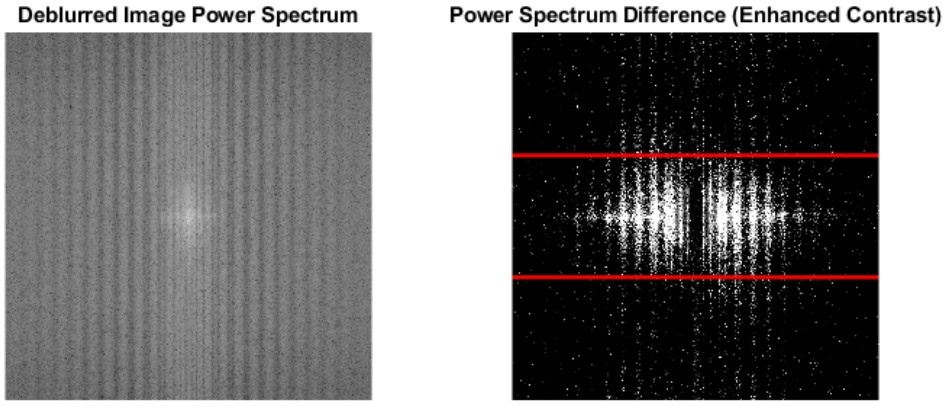

Figure 4.33: Deblurred image power spectra from simulated 1/60 second exposure time trajectory and power spectra difference from static image power spectra

Table 4.12: Exposure time simulated power spectra difference means μ_{PS}, standard deviations σ_{PS}, and t-test scores t_{PS} by trajectory over five trials

Trajectory	μ_{PS}	σ_{PS}	t_{PS}
15 FPS	7091.9	2.868	9.999E-01
30 FPS	7070.7	1.208	9.913E-01
45 FPS	7064.9	1.625	4.833E-01
60 FPS	7063.4	3.310	2.228E-01

Note: $*p < 0.05$ compared to linear trajectory

Table 4.13: Exposure time simulation power spectra score t-test p-values matrix by trajectory

Trajectory	15FPS	30FPS	45FPS	60FPS
15FPS	-	1	1	1
30FPS	7.717E-06*	-	1	0.992
45FPS	3.637E-06*	1.793E-04*	-	0.765
60FPS	2.037E-04*	7.862E-03*	0.235	-

Note: $*p < 0.05$

4.4 Discussion

The results of the simulation show an interesting trend; all of the RTDs for the RBF-based trajectories show either a single or multiple spikes in residence time at the center of the trajectory. This makes sense, as this translates to a blur kernel with the highest values very close to the center that generates less blur to begin with as well as one that would be easier and more predictable to deblur with. In an ideal case, as is true in the simulation, where the motion and blur kernel are perfectly known, Richardson-Lucy deconvolution would create a very close recreation of the original image. However, this is shown to be false as more linear-shaped trajectories failed to yield useful results. The single peak RTD maximizes the residence time around the center of the image, while the multi-peak RTD is likely a consequence of a less ideal optimization that could generally be achieved with a lower acceleration limit. This generates multiple phase distortions in the image that can be stitched together during the deblurring process which leads to strong artifacts, but still produces high contrast edges for text recognition. The concentration of residence time at the trajectory center is directly related to the overall OCR performance and indicates a possible attempt of saccade-like motion [39].

CHAPTER 5
EXPERIMENTATION

5.1 Experiment Setup

In order to verify the various results found in the optimization, an experimental setup was designed to replace the simulated capture process with a real, controlled-blur capture process. Figure 5.1 shows the setup used. It comprises a global shutter camera (model XiC, 2.3 MP CMOS, Ximea Co.) with a 12 mm fixed lens (model: M118FM12 C-Mount 12 mm, Tamron Co. Ltd., Saitama, Japan) attached to a 50 mm stroke voice coil stage (model: VCS24-029-LB-12, H2W Technologies, Inc.,CA, USA) controlled via a single axis motion controller with power supply (model: DMC-30012, Galil, California, USA). The motion controller is connected to a computer via Ethernet for socket messaging communication while the camera is connected via USB and are synchronized based on a timed delay between motion start and the beginning of image capture.

The voice coil type linear stage was chosen due to the high level of acceleration it can achieve as well as high speed and accuracy. To conduct a capture, a time-series of position points from one of the selected optimization trials is sent to the motion controller while the camera is timed to capture at the start of movement and finish at the end of motion. A trajectory width of 40 mm was chosen in order to allow some buffer space to prevent accidental collision with the ends of the linear stage. The camera was also placed at a distance of 0.785 m from the image in order to make the camera capture an image 0.5 m in width. Every test image used was printed with a width of 0.216 m. A studio light was placed behind the camera facing the image wall in order to ensure even lighting for all captures.

Trials were run to replicate the data obtained in the first and second simulation tests.

The third set could not be replicated because the voice coil was not capable of producing trajectories at that high velocity.

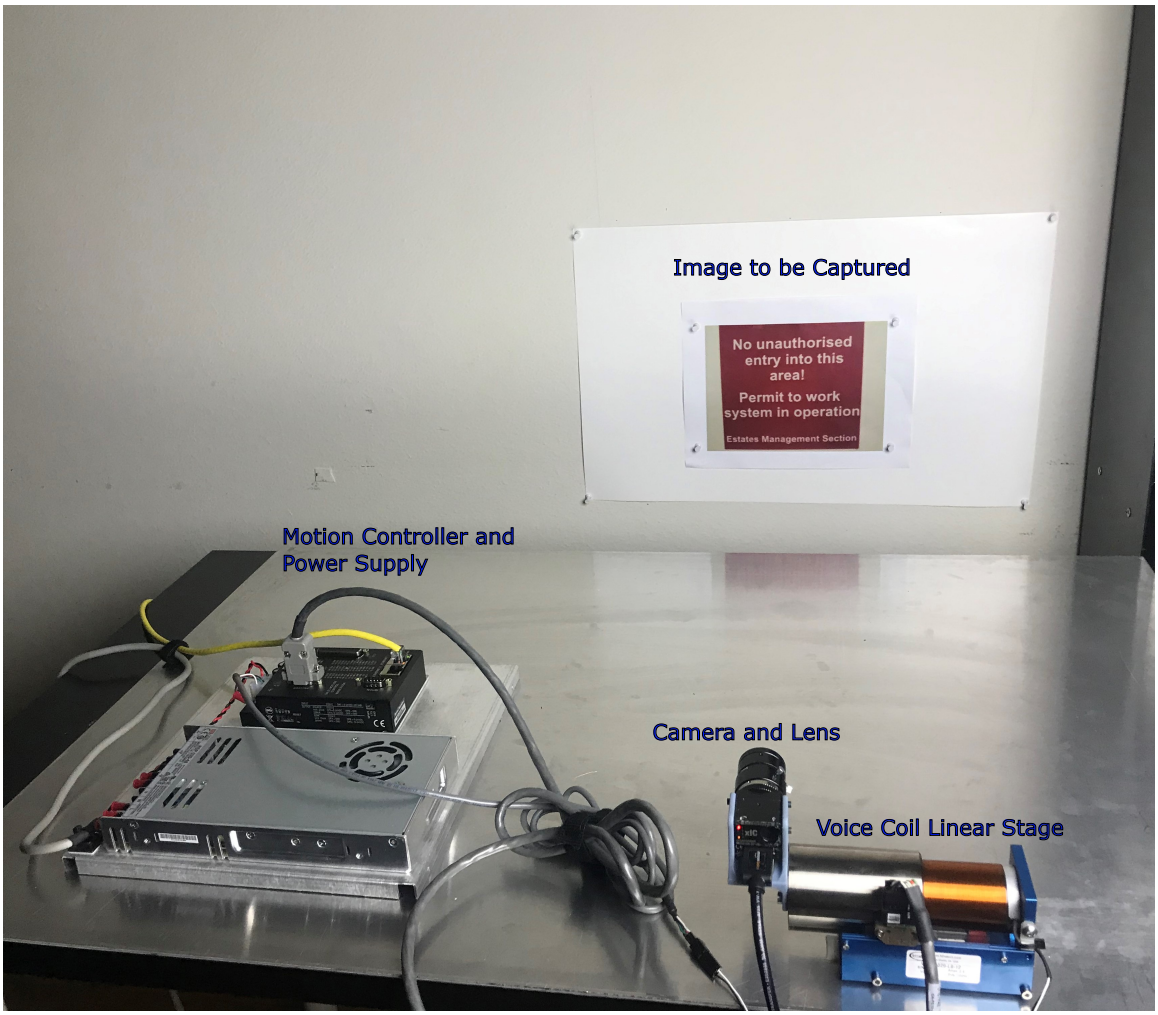

Figure 5.1: Experimental setup for the controlled-blur capture process

5.2 Experiment Results

For the first set of trials, samples from the first image in the image set are shown in Figures 5.2 to 5.8. These figures show the deblurred image and the attempted trajectory as well as the recorded trajectory.

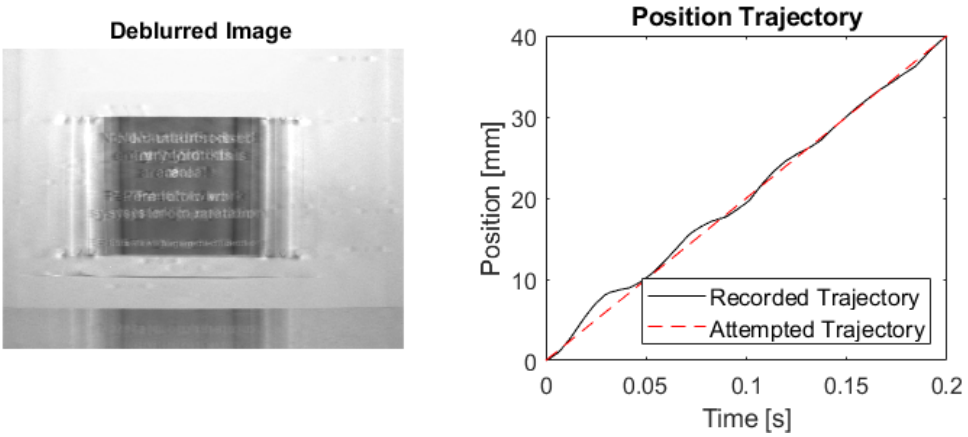

Figure 5.2: Sample of experiment deblurred image using linear trajectory with position, velocity, and residence time distribution plots

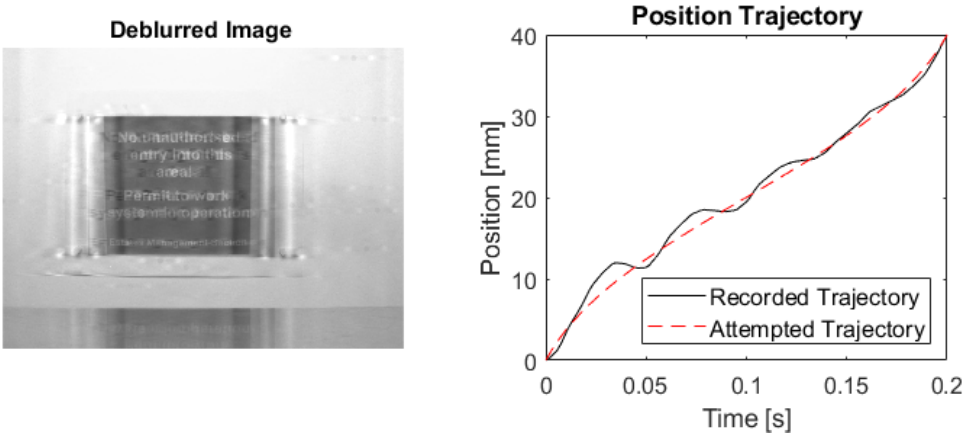

Figure 5.3: Sample of experiment deblurred image using inverse error trajectory with position, velocity, and residence time distribution plots

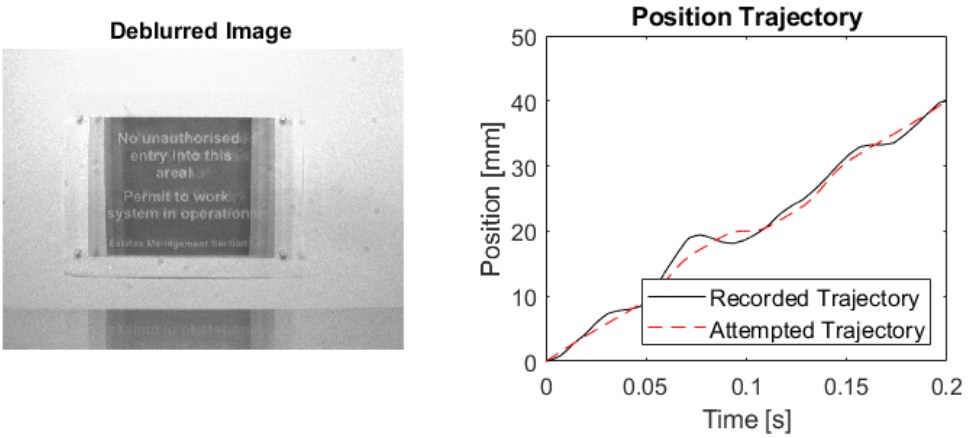

Figure 5.4: Sample of experiment deblurred image using RBF1 trajectory with position, velocity, and residence time distribution plots

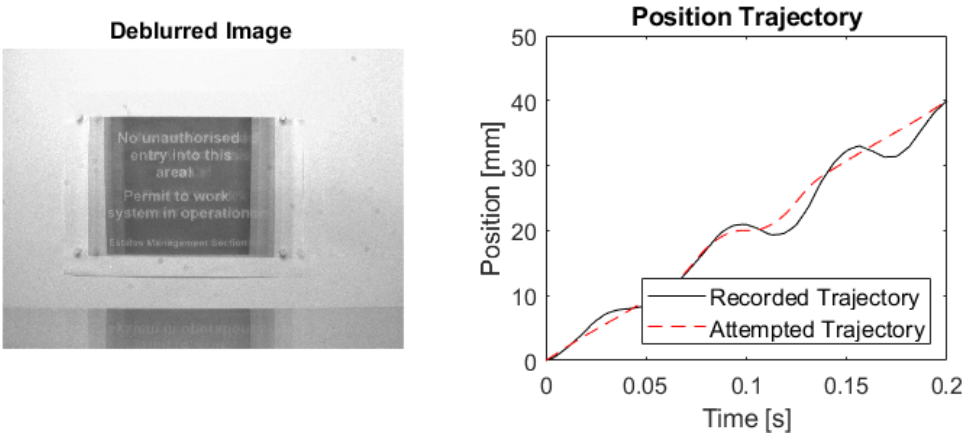

Figure 5.5: Sample of experiment deblurred image using RBF2 trajectory with position, velocity, and residence time distribution plots

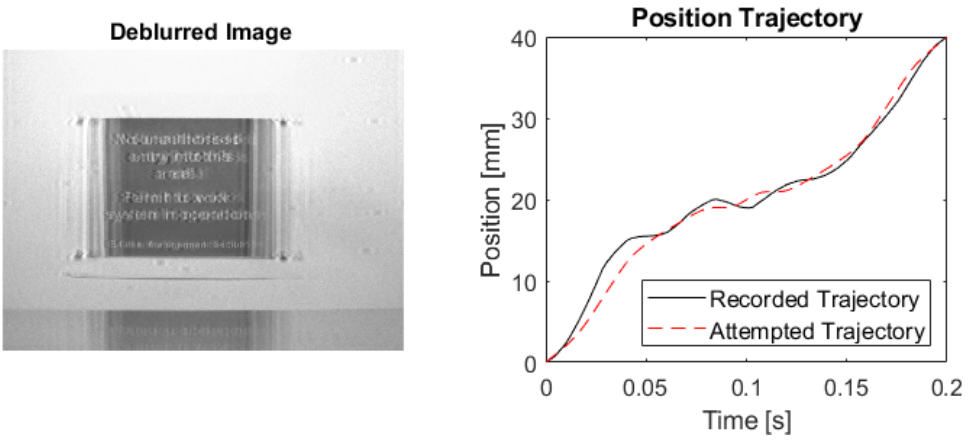

Figure 5.6: Sample of experiment deblurred image using RBF3 trajectory with position, velocity, and residence time distribution plots

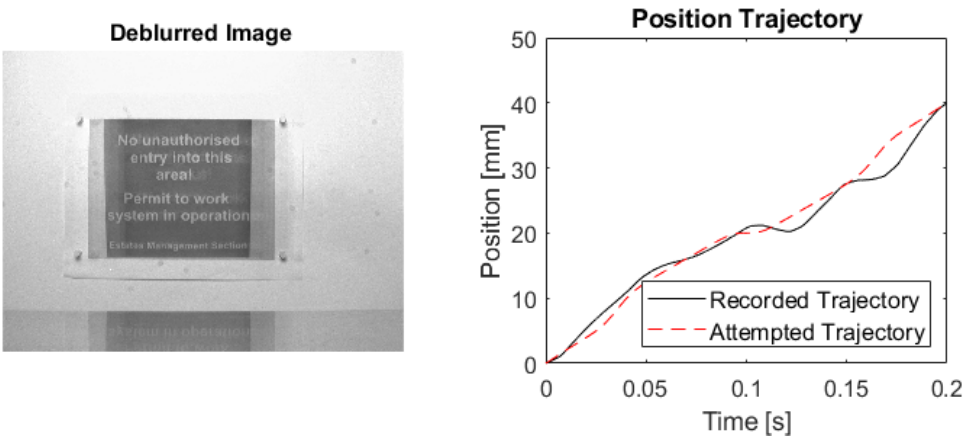

Figure 5.7: Sample of experiment deblurred image using RBF4 trajectory with position, velocity, and residence time distribution plots

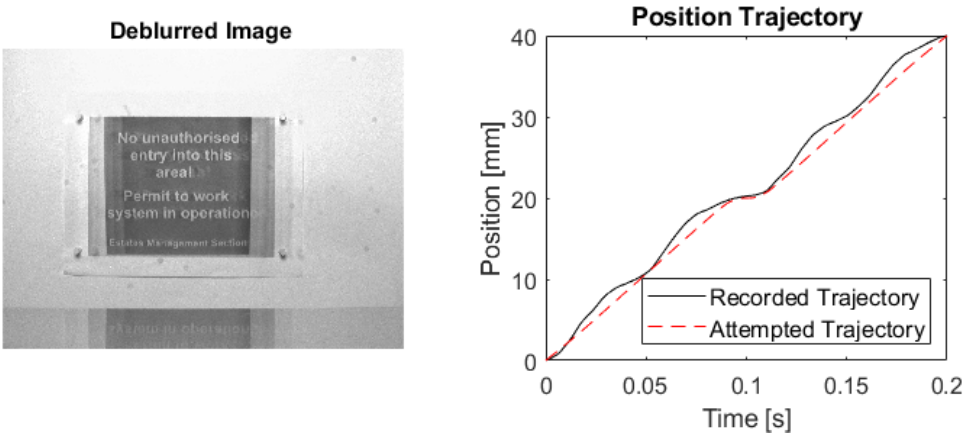

Figure 5.8: Sample of experiment deblurred image using RBF5 trajectory with position, velocity, and residence time distribution plots

Although the linear stage is supposedly following the simulated trajectory, one of the most noticeable observations of the sample images is that these images are significantly more blurred. This is likely due to the slight differences in the trajectories creating discrepancy with the blur kernel used in deconvolution and the actual blur kernel that is affecting the image capture. Table 5.1 details these discrepancies by calculating the cross correlation between the all trajectories and showing their statistical significance against either all attempted or all recorded trajectories. All but two values have (t_{rec} for Inverse and RBF2) show $p < 0.05$ statistical significance, showing that the linear stage is mostly capable of producing trajectories that are distinctly different from each other and are within an acceptable error range.

Table 5.1: Main search experiment recorded vs. attempted normalized cross correlation X with t-test scores vs. all attempted t_{att} and t-test scores vs. all recorded t_{rec}

Trajectory	X	t_{att}	t_{rec}
Linear	0.99941***	1.726E-02	7.593E-03
Inverse	0.99773*	3.203E-02	1.670E-01
RBF1	0.99898***	7.141E-03	3.513E-02
RBF2	0.99816*	2.715E-02	8.052E-02
RBF3	0.99839***	7.009E-05	5.277E-04
RBF4	0.99795***	2.899E-03	5.332E-03
RBF5	0.99956***	5.478E-03	5.856E-03

Notes: *$p < 0.05$ for t_{att}, **$p < 0.05$ for t_{rec}, ***$p < 0.05$ for both t_{att} and t_{rec}

Despite small variation from the planned path, the RBF trajectories still significantly outperformed the linear trajectory up to 29.4% in the case of best performing RBF4. The results for all trajectories are shown in Table 5.2.

Table 5.2: Main search experiment score means μ_{OCR}, standard deviations σ_{OCR}, and t-test scores t_{OCR} by trajectory

Trajectory	μ_{OCR}	σ_{OCR}	t_{OCR}
Linear	2.744	0.742	-
Inverse	2.776	0.647	5.738E-01
RBF1	2.045*	1.289	5.507E-03
RBF2	2.109*	1.290	8.775E-03
RBF3	2.314*	1.080	2.853E-02
RBF4	1.937*	1.340	6.406E-03
RBF5	2.035*	1.324	5.897E-03

Note: *$p < 0.05$ compared to linear trajectory

Similar to the simulated tests, the left-tailed t-test p-value matrix is calculated to compare relative performance between all trajectories in Table 5.3. In this case, there is no discernible difference in performance outside of the obvious improvement of all RBF trajectories over the linear and inverse error trajectories. Sample experiment score and OCR data for RBF4 can be found in Table A3.

Table 5.3: Main search experiment t-test p-values matrix by trajectory

Trajectory	Linear	Inverse	RBF1	RBF2	RBF3	RBF4	RBF5
Linear	-	0.426	0.994	0.991	0.971	0.994	0.994
Inverse	0.574	-	0.997	0.995	0.982	0.998	0.997
RBF1	5.507E-03*	3.012E-03*	-	0.349	0.136	0.656	0.520
RBF2	8.775E-03*	5.482E-03*	0.651	-	0.209	0.783	0.755
RBF3	2.853E-02*	0.018*	0.864	0.791	-	0.869	0.842
RBF4	6.406E-03*	1.582E-03*	0.344	0.217	0.131	-	0.301
RBF5	5.897E-03*	3.260E-03*	0.480	0.245	0.158	0.699	-

Note: *$p < 0.05$

Sample images for the acceleration limit experiments and their respective trajectory plots are shown in Figures 5.9 to 5.12.

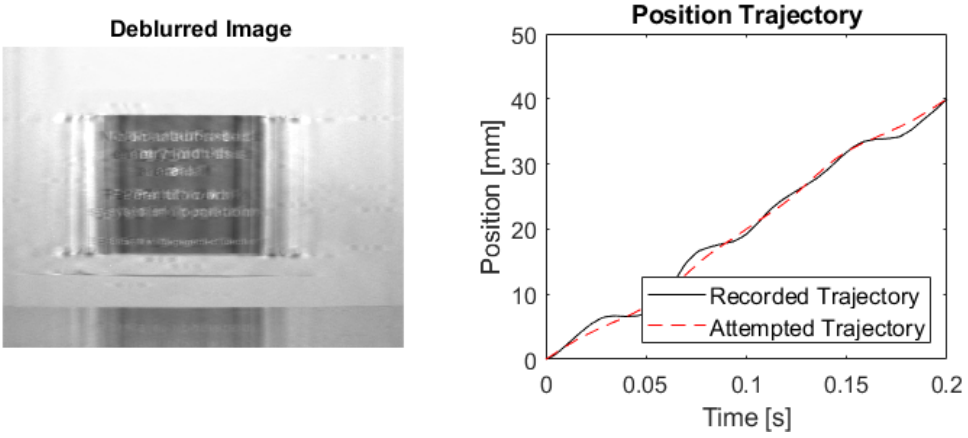

Figure 5.9: Sample of experiment deblurred image using an optimized trajectory for a $1g$ acceleration limit with position, velocity, and residence time distribution plots

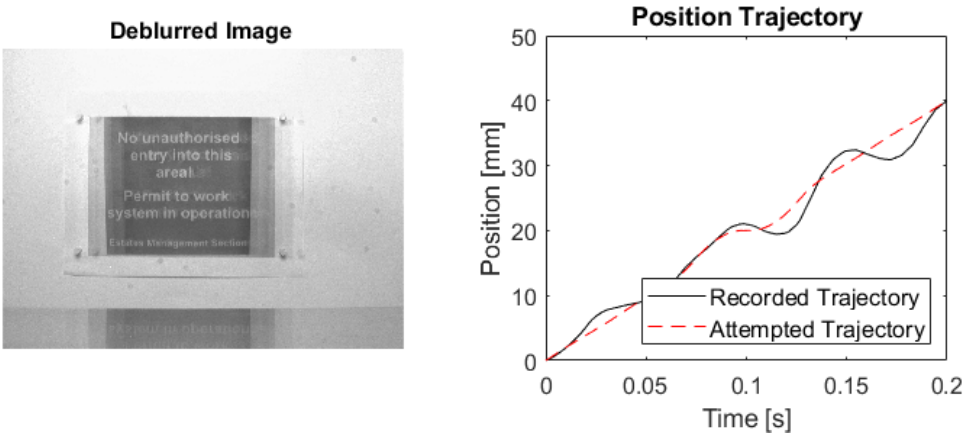

Figure 5.10: Sample of experiment deblurred image using an optimized trajectory for a $2g$ acceleration limit with position, velocity, and residence time distribution plots

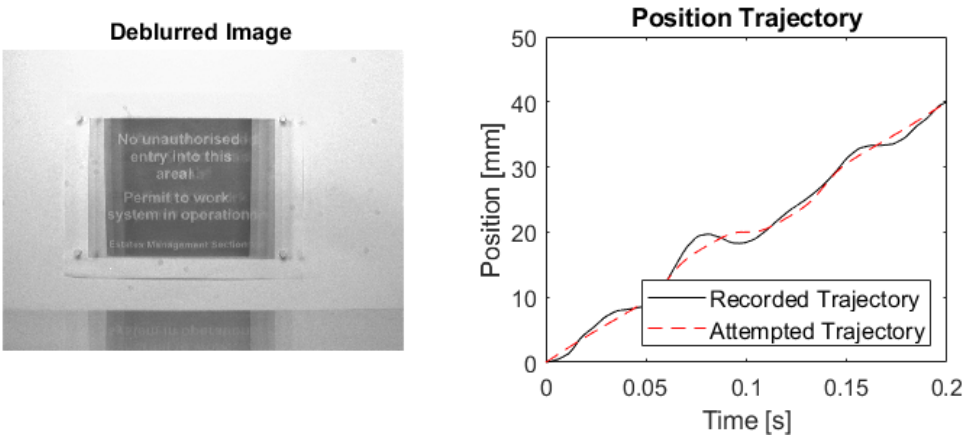

Figure 5.11: Sample of experiment deblurred image using an optimized trajectory for a $3g$ acceleration limit with position, velocity, and residence time distribution plots

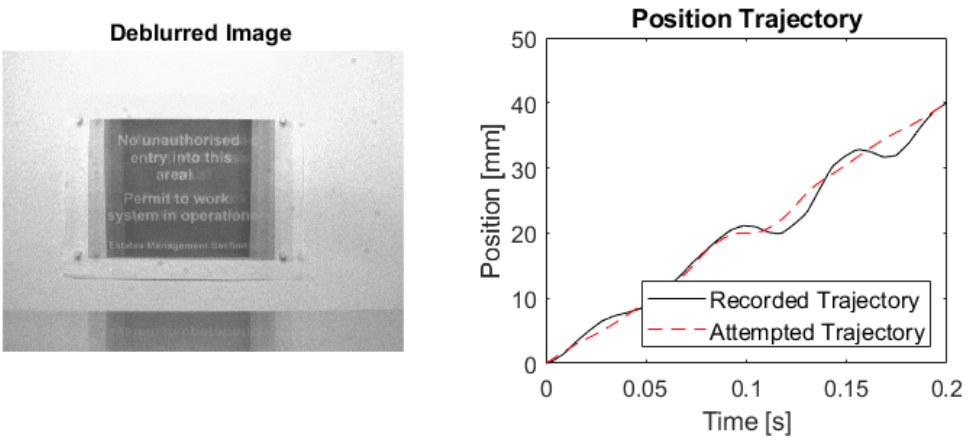

Figure 5.12: Sample of experiment deblurred image using an optimized trajectory for a $4g$ acceleration limit with position, velocity, and residence time distribution plots

Once again, the cross correlation and statistical significance of each trajectory set is calculated in Table 5.4. Only one value here is not statistically significant (t_{acc} for the $2g$ limit trajectory). This reaffirms the ability of the experimental setup to recreate the simulated trajectories.

Table 5.4: Acceleration limit experiment recorded vs. attempted normalized cross correlation X with t-test scores vs. all attempted t_{att} and t-test scores vs. all recorded t_{rec}

Trajectory	X	t_{att}	t_{rec}
$1g$ Limit	0.99917***	1.008E-02	2.584E-02
$2g$ Limit	0.99847**	5.397E-02	4.987E-02
$3g$ Limit	0.99951***	1.174E-02	7.852E-03
$4g$ Limit	0.99907***	3.021E-05	3.320E-04
$5g$ Limit	0.99795***	2.899E-03	5.332E-03

Notes: *$p < 0.05$ for t_{att}, **$p < 0.05$ for t_{rec}, ***$p < 0.05$ for both t_{att} and t_{rec}

The results in this series of tests very closely match those of the simulated series. In fact, a direct relationship between acceleration limit and OCR performance becomes more clear here as shown in Table 5.5. All of the trajectories showed statistically significant improvement against the linear trajectory except for the $1g$ limit trajectory. It is interesting to note that due to the lower acceleration, the voice coil linear stage was better able to replicate the attempted trajectory, yet this still resulted in worse overall performance.

Table 5.5: Acceleration limit experiment score means μ_{OCR}, standard deviations σ_{OCR}, and t-test scores t_{OCR} by trajectory

Trajectory	μ_{OCR}	σ_{OCR}	t_{OCR}
$1g$ Limit	2.857	0.516	7.676E-01
$2g$ Limit	2.174*	1.308	1.629E-02
$3g$ Limit	2.138*	1.266	3.666E-03
$4g$ Limit	2.061*	1.267	5.837E-03
$5g$ Limit	1.937*	1.340	6.406E-03

Note: *$p < 0.05$ compared to linear trajectory

Table 5.6 confirms the degraded performance of the $1g$ case while also showing no difference in the remaining cases via p-value comparison.

Table 5.6: Acceleration limit experiment t-test p-values matrix by trajectory

Trajectory	$1g$	$2g$	$3g$	$4g$	$5g$
$1g$	-	0.998	0.999	0.999	1
$2g$	2.434E-03*	-	0.570	0.758	0.838
$3g$	1.197E-03*	0.430	-	0.644	0.812
$4g$	5.229E-04*	0.242	0.356	-	0.707
$5g$	2.877E-04*	0.162	0.188	0.293	-

Note: *$p < 0.05$

5.3 Spectral Analysis

As with the simulated results, the experimental results were analyzed using spectral analysis. The static experiment image for image 1 and its power spectra is shown in Figure 5.13. The power spectra for the same sample image and difference from the static image are shown in Figures 5.14 to 5.20 for the linear, inverse error, and RBF trajectories 1-5.

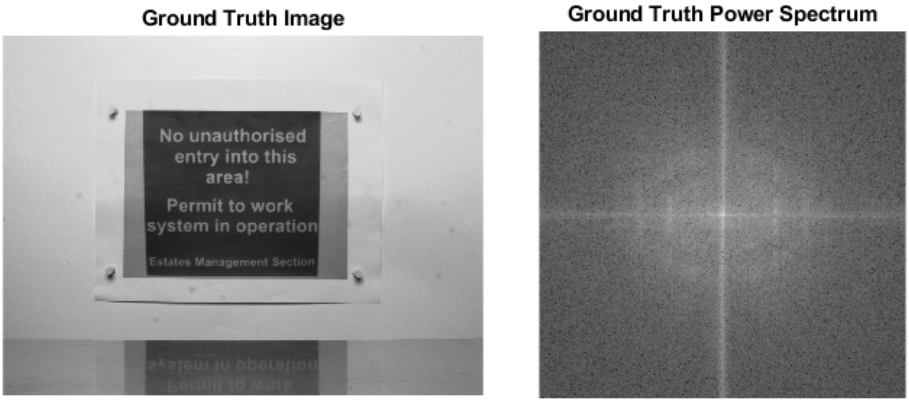

Figure 5.13: Static experiment image 1 and its power spectra

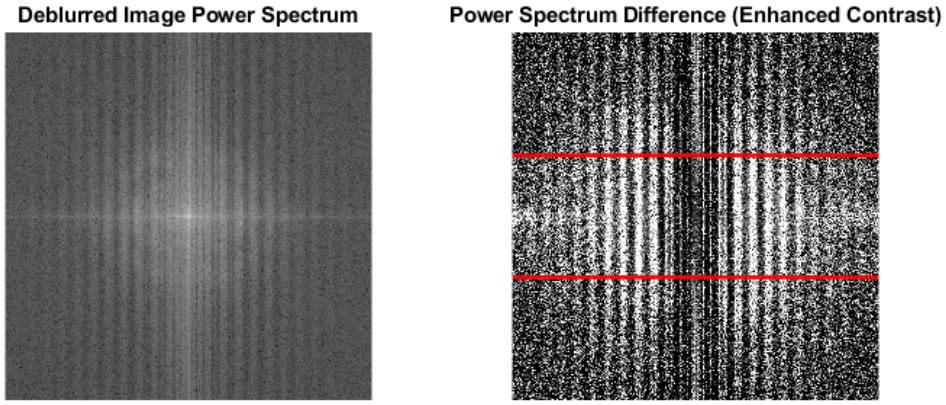

Figure 5.14: Deblurred image power spectra from experiment linear trajectory and power spectra difference from static image power spectra

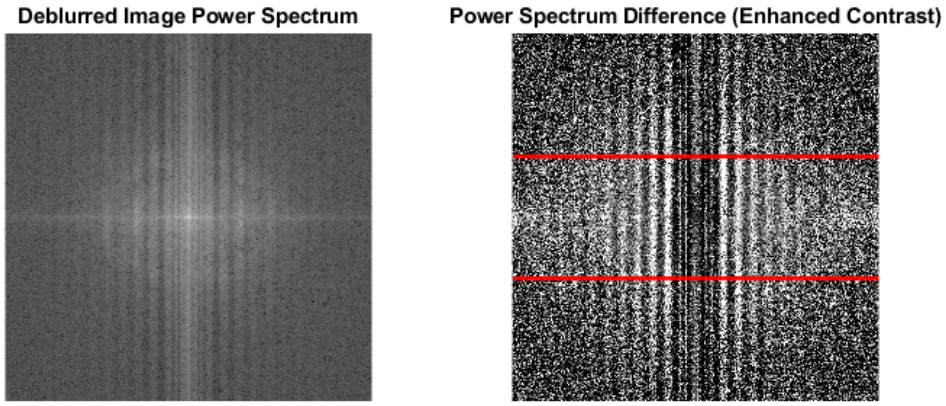

Figure 5.15: Deblurred image power spectra from experiment inverse error trajectory and power spectra difference from static image power spectra

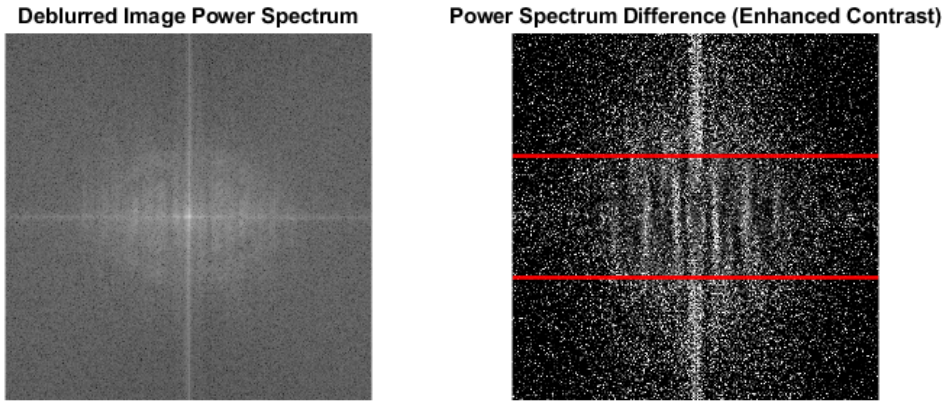

Figure 5.16: Deblurred image power spectra from experiment RBF1 and power spectra difference from static image power spectra

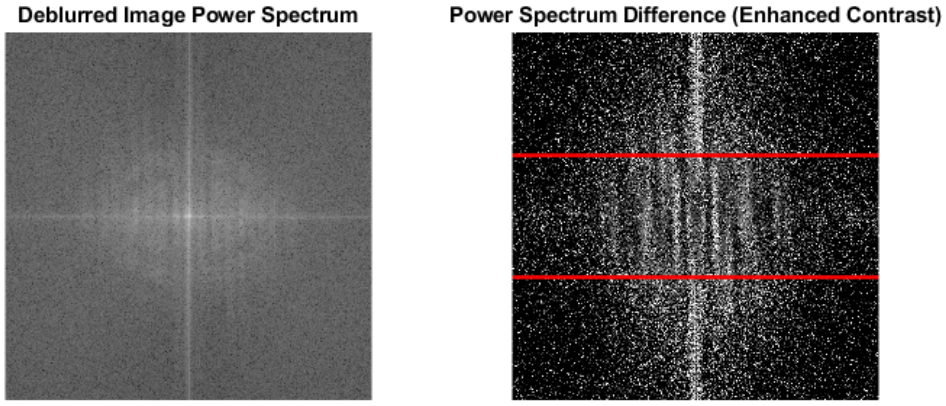

Figure 5.17: Deblurred image power spectra from experiment RBF2 and power spectra difference from static image power spectra

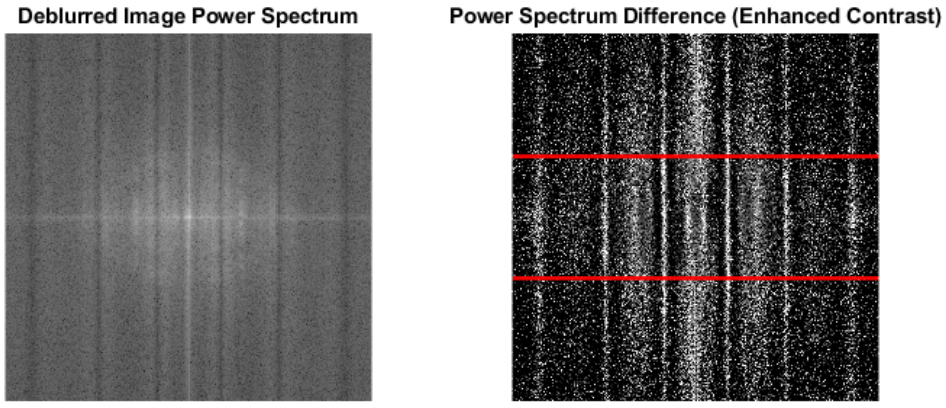

Figure 5.18: Deblurred image power spectra from experiment RBF3 and power spectra difference from static image power spectra

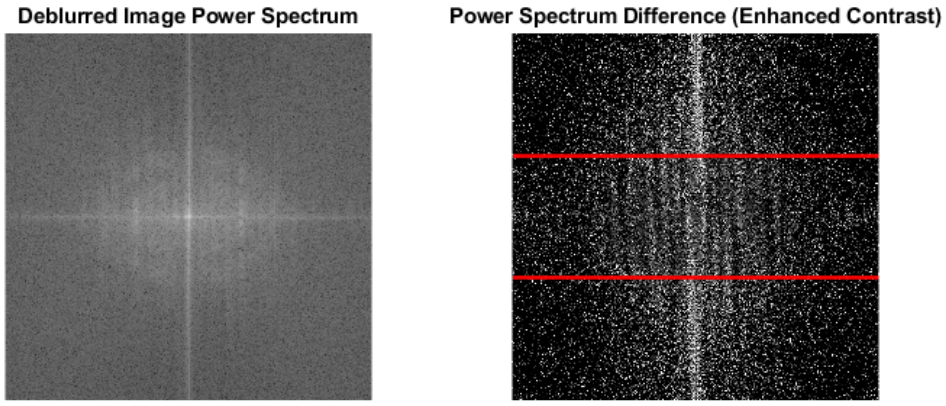

Figure 5.19: Deblurred image power spectra from experiment RBF4 and power spectra difference from static image power spectra

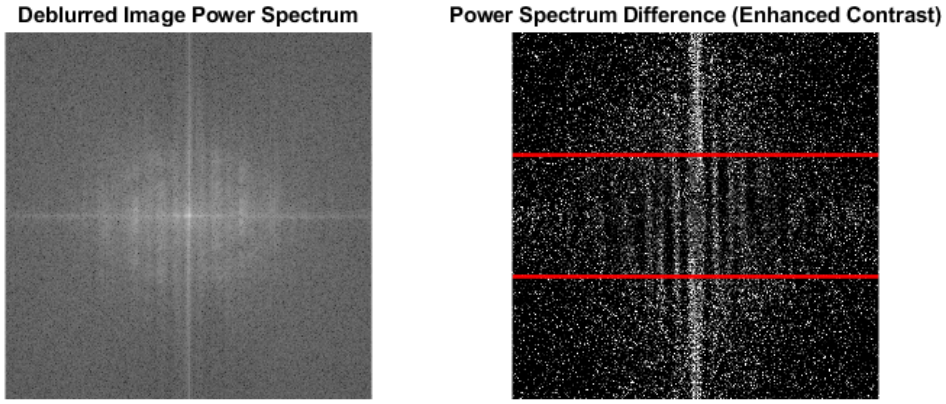

Figure 5.20: Deblurred image power spectra from experiment RBF5 and power spectra difference from static image power spectra

The decreased deblurring performance is immediately noticeable across all of the power spectra with the increased presence of vertical bands that are produced by ringing artifacts in the deblurred images. That being said, all of the RBF trajectories still outperform the linear trajectory by a significant margin as shown in Table 5.7. There is a less distinctive loss in purely horizontal components in these power spectra as the original image has less relative horizontal high frequency information, likely due to the smaller text in the frame and lower contrast due to the real capture process. This explains why the loss in the difference spectrum is more distributed around lower frequencies. The p-values in Table 5.8

show under-performance by RBF3, which matches the OCR results.

Table 5.7: Main search experiment power spectra difference means μ_{PS}, standard deviations σ_{PS}, and t-test scores t_{PS} by trajectory

Trajectory	μ_{PS}	σ_{PS}	t_{PS}
Linear	8419.5	8837.7	-
Inverse	7192.1	1786.9	2.288E-01
RBF1	4548.8*	824.1	1.554E-02
RBF2	4583.4*	840.4	1.576E-02
RBF3	4995.7*	944.4	2.796E-02
RBF4	4562.9*	947.6	1.564E-02
RBF5	4476.8*	500.8	1.366E-02

Note: *$p < 0.05$ compared to linear trajectory

Table 5.8: Main search experiment power spectra score t-test p-values matrix by trajectory

Trajectory	Linear	Inverse	RBF1	RBF2	RBF3	RBF4	RBF5
Linear	-	0.771	0.984	0.984	0.972	0.984	0.986
Inverse	0.229	-	1	1	1	1	1
RBF1	0.016*	6.709E-09*	-	0.414	0.015*	0.469	0.739
RBF2	0.016*	3.677E-08*	0.586	-	2.513E-04*	0.552	0.789
RBF3	0.028*	1.045E-06*	0.985	1	-	0.985	0.998
RBF4	0.016*	5.847E-09*	0.531	0.448	0.015*	-	0.709
RBF5	0.014*	6.487E-09*	0.261	0.211	2.209E-03*	0.291	-

Note: *$p < 0.05$

Spectral analysis results for the experimental acceleration limit tests are shown in Figures 5.21 to 5.24.

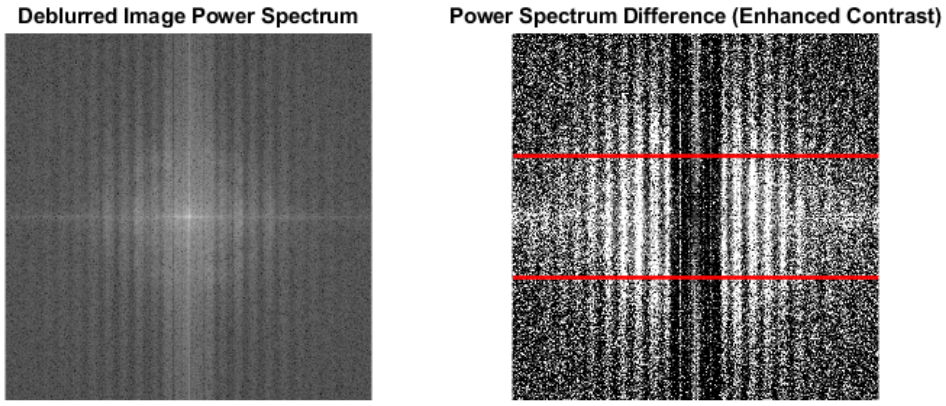

Figure 5.21: Deblurred image power spectra from experiment $1g$ acceleration limit trajectory and power spectra difference from static image power spectra

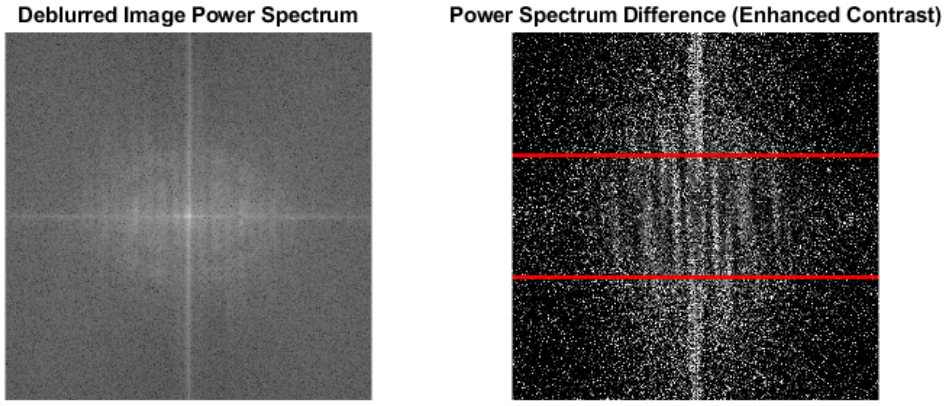

Figure 5.22: Deblurred image power spectra from experiment $2g$ acceleration limit trajectory and power spectra difference from static image power spectra

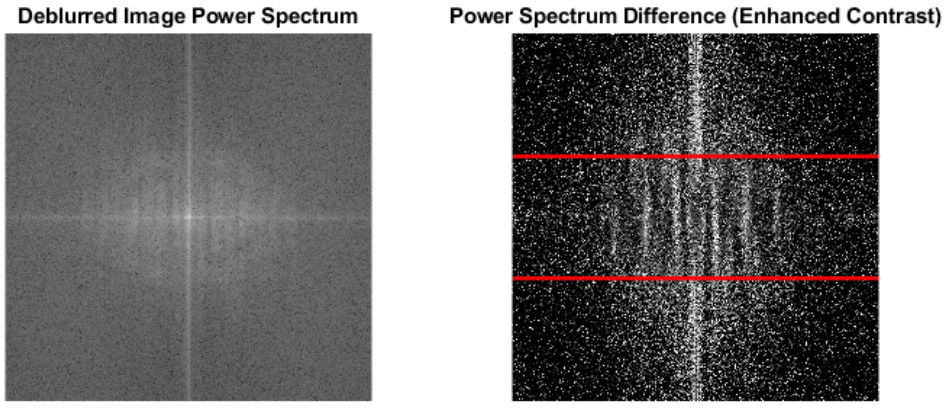

Figure 5.23: Deblurred image power spectra from experiment $3g$ acceleration limit trajectory and power spectra difference from static image power spectra

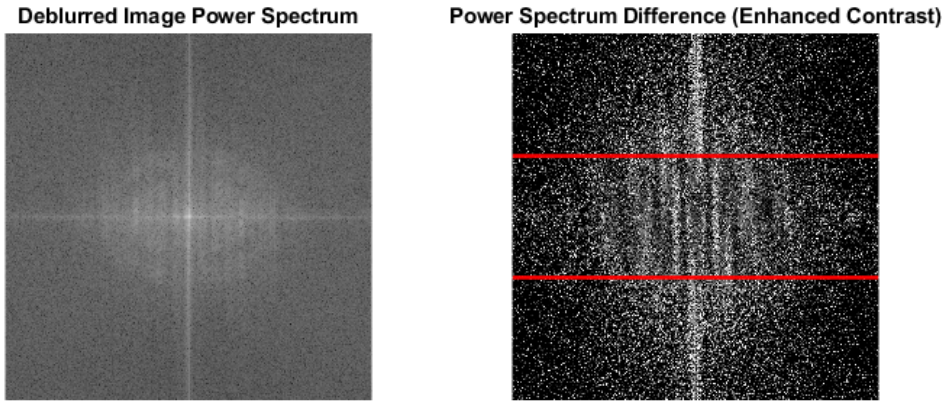

Figure 5.24: Deblurred image power spectra from experiment $4g$ acceleration limit trajectory and power spectra difference from static image power spectra

The results here confirm the results seen in the simulated deblurring, simulated power spectra, and the experimental deblurring results. The $2g$, $3g$, and $4g$ spectra all look and score similarly as calculated in Table 5.9, while the $1g$ case shows significantly greater intensity loss in both higher and lower frequency horizontal components with confirmed statistical significance shown in Table 5.10.

Table 5.9: Acceleration limit experiment power spectra difference means μ_{PS}, standard deviations σ_{PS}, and t-test scores t_{PS} by trajectory

Trajectory	μ_{PS}	σ_{PS}	t_{PS}
$1g$ Limit	7114.2	1634.9	2.136E-01
$2g$ Limit	4751.0*	884.0	1.712E-02
$3g$ Limit	4791.8*	1004.3	1.804E-02
$4g$ Limit	4665.9*	1080.4	1.913E-02
$5g$ Limit	4562.9*	947.6	1.564E-02

Note: *$p < 0.05$ compared to linear trajectory

Table 5.10: Acceleration limit experiment power spectra score t-test p-values matrix by trajectory

Trajectory	$1g$	$2g$	$3g$	$4g$	$5g$
$1g$	-	1	1	1	1
$2g$	2.038E-10*	-	0.408	0.689	0.850
$3g$	6.114E-08*	0.592	-	0.721	0.851
$4g$	7.744E-11*	0.311	0.279	-	0.746
$5g$	1.899E-09*	0.150	0.149	0.254	-

Note: $*p < 0.05$

5.4 Discussion

The goal of the experimental tests was to verify and confirm the results found in the simulation and they do in fact reach these conclusions. The controlled capture process models a real-world application of motion planning for dynamics-based deblurring and introduces all of the difficulties of actual implementation. Lower contrast images and smaller text contribute to more challenging OCR. In addition, the minor deviation from the planned trajectory creates issues not only by creating variation from the optimal trajectory, but also by creating discrepancy between the true blur kernel and the blur kernel used for deblurring. That being said, the RBF trajectories still perform with statistically significant improvement compared to the linear and inverse error trajectories. The best scoring RBF trajectory, RBF4, yielded improvements of up to 29.4% with these improvements also showing in the spectral analysis. One key difference between the real and simulated trials is the difference in frequency components present in the power spectra. Examining the static images and their power spectra, the real capture shows much greater higher frequency radial components than the simulated image, likely due to the corners and edges of all text being weaker in the image. This has corresponding effects in the deblurred power spectra and the power

spectra differences, contributing to the weaker edge preservation.

CHAPTER 6
CONCLUDING REMARKS

6.1 Conclusion

This thesis has identified the need for investigation into motion planning for deblurring natural text images and proposed a novel optimization method for determining ideal, endpoint constrained trajectories for maximizing optical character recognition rates. RBF networks provide the ability to automatically generate smooth functions with a high degree of variability while being controlled by a relatively small set of tunable parameters that can be easily associated to the character recognition-based output score. It was found that trajectories with residence time distributions consisting of a single or multiple peaks performed best at maintaining edge features while still meeting all physical constraints. Trajectories fitting this general shape require a threshold of allowable acceleration in order to be achieved and falling below this value significantly impacts performance. While theoretically any trajectory in non-blind motion deblurring with a perfectly modeled blur kernel should be able to reproduce high accuracy deblurred images, this has been shown to not be the case potentially due to discretization introducing error into the deblurring process. For this reason, it can be said that high peak residence time distributions contribute to edge preservation and ultimately text recognition both by preventing image blur in the first place as well as by providing a well modeled blur kernel. The spectral analysis performed on the trajectory results confirm that RBF derived trajectories more effectively maintain text features by maintaining the high frequency components perpendicular to the blur direction.

The contributions of this thesis can be summarized in the list below.

- A novel optimization method for one-dimensional, dynamics-based motion deblurring using radial basis function network trajectory planning is developed specifically

with the goal of maximizing the recognition rates of text in natural images. Simulations and real-world experimental testing of these planned trajectories show significantly improved text recognition rates, validating the use of the optimization algorithm.

- Parameter constrained trajectories found using the optimization are identified and compared to analogous linear trajectories to show significant improvement. Features of the residence time distribution are identified that explain why certain trajectories are ideal for motion deblurring.

- Spectral analysis with an emphasis on edge preservation in text is implemented as a subsequent evaluation method of deblurred images from the optimized trajectories. A unique evaluation method of comparing the base image and deblurred image power spectra is related to the character recognition performance.

6.2 Future Work

Future work on this subject should focus on improving the overall text recognition rate using the proposed algorithm. Because the optimization developed is ambiguous to any particular method of image processing or deblurring algorithm, other unique approaches should be attempted in order to yield even greater improvement. The optimization can also be expanded to a two-dimensional or greater environment to find ideal trajectories for a wide variety of applications. Further work should also be done in real-world environments, such as on an autonomous vehicle, in order to further evaluate planned trajectories. In such cases, hardware platforms with varying capabilities, such as linear stage speed, should be tested to draw a clear relationship between maximum deblurring ability and hardware specifications. In addition, factors like camera resolution would affect the blur kernel size and accuracy and contribute to deblurring performance as well.

Appendices

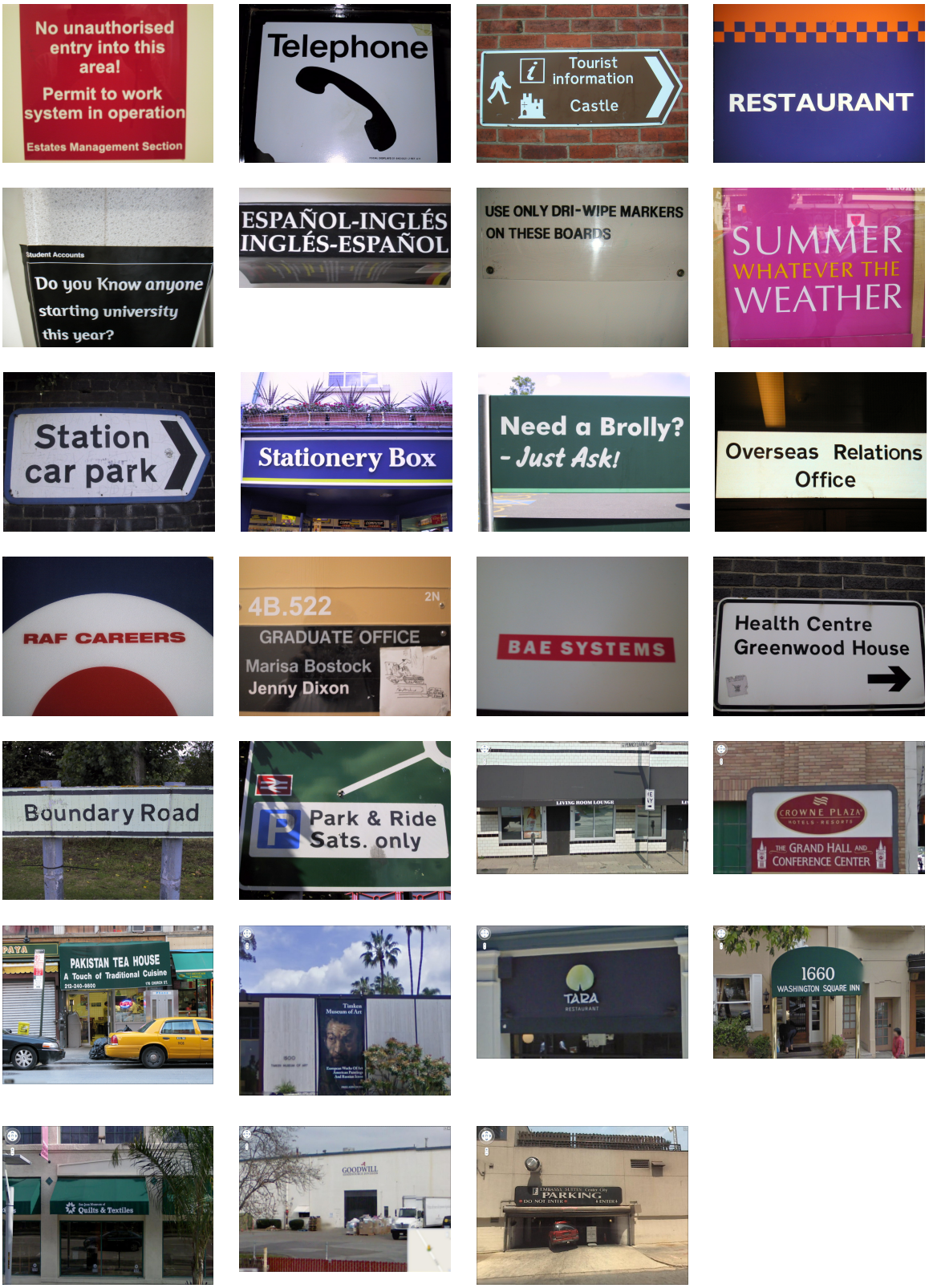

Figure A1: Image set used by the optimization consisting of score filtered images from the ICDAR 2015 competition and Street View Text Dataset

Table A1: Image set ground truth text and mean static OCR score

Image Set Number	Mean Static OCR Score	Ground Truth OCR Text
1	0.068	NounauthorizedentryintothisareaPermittoworksysteminoperationEstatesManagementSection
2	0.111	Telephone
3	0.492	TouristinformationCastle
4	0	RESTAURANT
5	0.453	StudentAccountsDoyouknowanyonestartinguniversitythisyear?
6	0.341	ESPANOL-INGLESINGLESESPANOL
7	0.269	USEONLYDRI-WIPEMARKERSONTHESEBOARDS
8	0.392	SUMMERWHATEVERTHEWEATHER
9	0.186	Stationcarpark
10	0.150	StationaryBox
11	0.038	NeedaBrolly?-JustAsk!
12	0	OverseasRelationsOffice
13	0	RAFCAREERS
14	0.489	4B.5222NGRADUATEOFFICEMarisaBostockJennyDixon
15	0.020	BAESYSTEMS
16	0	HealthCentreGreenwoodHouse
17	0.383	BoundaryRoad
18	0.420	PPark&RideSats.only
19	0.497	PENNSYLVANIALIVINGROOMLOUNGE
20	0.448	CROWNEPLAZATHEGRANDHALLANDCONFERENCECENTER
21	0.436	PAKISTANTEAHOUSEATouchofTraditionalCuisine212-240-9800176CHURCHST.
22	0.478	TimkenMuseumofArtEuropeanWorksOfArtAmericanPaintingsAndRussionIcons
23	0.200	TARARESTAURANT
24	0.311	1660WASHINGTONSQUAREINN
25	0.280	SanJoseMuseumofQuilts&Textiles
26	0.400	GOODWILL
27	0.135	EEMBASSYSUITESCenterCityPARKINGDONOTENTERENTER

Table A2: Sample simulation scores for RBF4 with trial 1 OCR text by image

Image Set Number	Mean RBF4 OCR Score	Trial 1 OCR Text
1	0.064	NounauthorisedentryintothisarealPermittoworksysteminoperationEstatesManagementSection
2	3	NO TEXT RECOGNIZED
3	0.440	faTourist,R”Ctormationia;Castle
4	0	RESTAURANT
5	1.846	StudentAccounts\DoyouKnowanyone\\startinguniversity\thisyear?
6	0.200	ESPANOL-INGLESINCE]Bstowotc}DNN(O)
7	3	NO TEXT RECOGNIZED
8	0.798	WHATEV
9	0.233	StationcarPoke4
10	1.514	|StationeryBox
11	0.128	|NeedaBrolly?EeJustAsk!
12	0.017	OverseasRelationsOffice
13	0	RAFCAREERS
14	0.141	∼GRADUATEOFFICE*MarisaBostockfiJennyDixon—Kiscats
15	2.584	NO TEXT RECOGNIZED
16	0	HealthCentreGreenwoodHouse
17	0.703	BoundaryRoad—
18	3	NO TEXT RECOGNIZED
19	1.462	LIVINGROOMLOUNGELYi|i
20	1.010	NO TEXT RECOGNIZED
21	0.155	PAKISTANTEAHOUSEATouchofTraditionalCuisine|242-240-9800:isrecancn
22	0.284	TimkenMuseumofArtEarapeanWorksOFAntAnsericanPaiety:|
23	3	NO TEXT RECOGNIZED
24	2.538	reomeerrmeni
25	1.493	Sealemels€Ouilts&Textiles:ae
26	3	NO TEXT RECOGNIZED
27	0.342	BARKING”=DONOTENTER>4ENTERS

Table A3: Sample experiment scores for RBF4 with OCR text by image

Image Set Number	RBF4 OCR Score	OCR Text
1	0.789	UiaksleacalelabcisieAlia'miaicomdali:rea!
2	3	NO TEXT RECOGNIZED
3	0.254	Touristfoldaatehucele
4	0	RESTAURANT
5	0.471	apAccstudent
6	0.122	naeESPANOL-INGLESINGLES-ESPANOI
7	3	NO TEXT RECOGNIZED
8	3	NO TEXT RECOGNIZED
9	3	NO TEXT RECOGNIZED
10	3	NO TEXT RECOGNIZED
11	0	NeedaBrolly?-JustAsk!
12	0	OverseasRelationsOffice
13	3	NO TEXT RECOGNIZED
14	3	NO TEXT RECOGNIZED
15	3	NO TEXT RECOGNIZED
16	0.885	PeGe
17	1.490	5eaaeancreaneeniaceaiattatatenit
18	3	NO TEXT RECOGNIZED
19	0.251	LIVINGROOMLOUNGEin
20	3	NO TEXT RECOGNIZED
21	3	NO TEXT RECOGNIZED
22	3	NO TEXT RECOGNIZED
23	3	NO TEXT RECOGNIZED
24	0.030	1660WASHINGTON.SQUAREINN
25	3	NO TEXT RECOGNIZED
26	3	NO TEXT RECOGNIZED
27	3	NO TEXT RECOGNIZED

REFERENCES

[1] y. Balaji, M. B. Kumar, and y. Sujatha, "Text information extraction and analysis for autonomous vehicle," in *2017 IEEE International Conference on Signal Processing, Informatics, Communication and Energy Systems (SPICES)*, IEEE, 2017, pp. 1–6.

[2] M. Ben-Ezra and S. Nayar, "Motion-based motion deblurring," *IEEE transactions on pattern analysis and machine intelligence*, vol. 26, pp. 689–98, 2004.

[3] S. Yadav, C. Jain, and A. Chugh, "Evaluation of image deblurring techniques," *International Journal of Computer Applications*, vol. 139, pp. 32–36, Apr. 2016.

[4] Q. Shan, W. Xiong, and J. Jia, "Rotational motion deblurring of a rigid object from a single image," in *2007 IEEE 11th International Conference on Computer Vision*, IEEE, 2007, pp. 1–8.

[5] R. Fergus, B. Singh, A. Hertzmann, S. T. Roweis, and W. T. Freeman, "Removing camera shake from a single photograph," in *ACM SIGGRAPH 2006 Papers*, Association for Computing Machinery, 2006, pp. 787–794.

[6] H.-L. Yang, Y.-H. Chiao, P.-H. Huang, and S.-H. Lai, "Blind image deblurring with modified richardson-lucy deconvolution for ringing artifact suppression," in *Advances in Image and Video Technology*, Y.-S. Ho, Ed., Springer Berlin Heidelberg, 2012, pp. 240–251.

[7] S. Vasu, V. R. Maligireddy, and A. N. Rajagopalan, "Non-blind deblurring: Handling kernel uncertainty with cnns," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, 2018, pp. 3272–3281.

[8] J. Greenhalgh and M. Mirmehdi, "Detection and recognition of painted road surface markings," in *4th International Conference on Pattern Recognition Applications and Methods*, vol. 1, Jan. 2015.

[9] A. Agrawal and R. Raskar, "Optimal single image capture for motion deblurring," in *IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, 2009, pp. 2560–2567.

[10] G. Boracchi and A. Foi, "Modeling the performance of image restoration from motion blur," *IEEE Transactions on Image Processing*, vol. 21, no. 8, pp. 3502–3517, 2012.

[11] E. Kurimo, L. Lepistö, J. Nikkanen, J. Grén, I. Kunttu, and J. Laaksonen, "The effect of motion blur and signal noise on image quality in low light imaging," in *Image*

Analysis. SCIA, A.-B. Salberg, J. Y. Hardeberg, and R. Jenssen, Eds., Springer Berlin Heidelberg, 2009, pp. 81–90.

[12] R. Katoch and J. Ueda, "Edge-preserving camera trajectories for improved optical character recognition on static scenes with text," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 4467–4474, 2019.

[13] S.-H. Jung and C. Taylor, "Camera trajectory estimation using inertial sensor measurements and structure from motion results," in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, vol. 2, IEEE, 2001, pp. II–II.

[14] Y.-W. Tai, P. Tan, and M. S. Brown, "Richardson-lucy deblurring for scenes under a projective motion path," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1603–1618, 2011.

[15] R. Katoch, B. Fusaro, and J. Ueda, "Inverse error function trajectories for image reconstruction," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, 2018, pp. 7527–7532.

[16] S. Schuon and K. Diepold, "Comparison of motion de-blur algorithms and real world deployment," *Acta Astronautica*, vol. 64, no. 11, pp. 1050–1065, 2009.

[17] C. Cai, A. Liu, and B. Zhang, "Motion deblurring from a single image," in *2016 IEEE 20th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*, IEEE, 2016, pp. 406–410.

[18] H.-L. Yang, P.-H. Huang, and S.-H. Lai, "A novel gradient attenuation richardson–lucy algorithm for image motion deblurring," *Signal Processing*, vol. 103, pp. 399–414, 2014, Image Restoration and Enhancement: Recent Advances and Applications.

[19] A. Mosleh, P. Green, E. Onzon, I. Begin, and J. P. Langlois, "Camera intrinsic blur kernel estimation: A reliable framework," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, 2015, pp. 4961–4968.

[20] H.-M. Gutmann, "A radial basis function method for global optimization," *Journal of Global Optimization*, vol. 19, Jan. 2000.

[21] H. Mirinejad and T. Inanc, "A radial basis function method for direct trajectory optimization," in *2015 American Control Conference*, IEEE, 2015, pp. 4923–4928.

[22] R. Smith, "An overview of the tesseract ocr engine," in *Ninth International Conference on Document Analysis and Recognition*, vol. 2, IEEE, 2007, pp. 629–633.

[23] M. G. Marne, P. R. Futane, S. B. Kolekar, A. D. Lakhadive, and S. K. Marathe, "Identification of optimal optical character recognition (ocr) engine for proposed system," in *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, IEEE, 2018, pp. 1–4.

[24] J. Greenhalgh and M. Mirmehdi, "Recognizing text-based traffic signs," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 3, pp. 1360–1369, 2015.

[25] M. R. Islam, C. Mondal, M. K. Azam, and A. S. M. J. Islam, "Text detection and recognition using enhanced mser detection and a novel ocr technique," in *2016 5th International Conference on Informatics, Electronics and Vision (ICIEV)*, IEEE, 2016, pp. 15–20.

[26] A. Dengel, F. Asad, A. Ul-Hasan, and F. Shafait, "High performance ocr for camera-captured blurred documents with lstm networks," in *DAS 2016, 12th Int'l IAPR Workshop on Document Analysis Systems*, IEEE, Apr. 2016.

[27] M. Saraswat and A. Sharma, "Genetic algorithm for optimization using matlab," *International Journal of Advanced Research in Computer Science*, vol. 4, 2013.

[28] J. Roberts, A. Cassula, J. Silveira, P. Prado, and J. Freire, "Gatoolbox: A matlab-based genetic algorithm toolbox for function optimization," Nov. 2017.

[29] S. El-Regaily, M. A. El-Aziz, H. El-Messiry, and M. Roushdy, "Using gpu-accelerated genetic algorithm for non-linear motion deblurring in a single image," in *2012 8th International Conference on Informatics and Systems (INFOS)*, IEEE, 2012, BIO-174-BIO–180.

[30] S. Su and W. Heidrich, "Rolling shutter motion deblurring," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, 2015, pp. 1529–1537.

[31] V. C. Kieu, F. Cloppet, and N. Vincent, "Bnrfbe method for blur estimation in document images," in *Advanced Concepts for Intelligent Vision Systems*, S. Battiato, J. Blanc-Talon, G. Gallo, W. Philips, D. Popescu, and P. Scheunders, Eds., Springer International Publishing, 2015, pp. 3–14.

[32] L. Yujian and L. Bo, "A normalized levenshtein distance metric," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 6, pp. 1091–1095, 2007.

[33] A. Verma and D. Singh, "Text deblurring using ocr word confidence," *International Journal of Image, Graphics and Signal Processing*, vol. 9, pp. 33–40, Jan. 2017.

[34] R. Millane, S. Alzaidi, and W.-H. Hsiao, "Scaling and power spectra of natural images," in *Image and Vision Computing NZ*, Jan. 2003.

[35] A. van der Schaaf and J. van Hateren, "Modelling the power spectra of natural images: Statistics and information," *Vision Research*, vol. 36, no. 17, pp. 2759–2770, 1996.

[36] M. K. Azadboni and A. Behrad, "Text detection and character extraction in color images using fft domain filtering and svm classification," in *6th International Symposium on Telecommunications (IST)*, IEEE, 2012, pp. 794–799.

[37] M. Narwaria, W. Lin, I. V. McLoughlin, S. Emmanuel, and L.-T. Chia, "Fourier transform-based scalable image quality measure," *IEEE Transactions on Image Processing*, vol. 21, no. 8, pp. 3364–3377, 2012.

[38] M. Ebrahimi Moghaddam and M. Jamzad, "Linear motion blur parameter estimation in noisy images using fuzzy sets and power spectrum," *EURASIP J. Adv. Sig. Proc.*, vol. 2007, Jan. 2007.

[39] M. D. Kim and J. Ueda, "Real-time panoramic image generation and motion deblurring by using dynamics-based robotic vision," *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 3, pp. 1376–1387, 2016.

[40] K. De and V. Masilamani, "Image sharpness measure for blurred images in frequency domain," *Procedia Engineering*, vol. 64, pp. 149–158, 2013, International Conference on Design and Manufacturing.

[41] S. Priambada and D. Widyantoro, "Levensthein distance as a post-process to improve the performance of ocr in written road signs," in *Second International Conference on Informatics and Computing*, IEEE, Nov. 2017, pp. 1–6.

[42] D. Karatzas, F. Shafait, S. Uchida, M. Iwamura, L. G. i. Bigorda, S. R. Mestre, J. Mas, D. F. Mota, J. A. Almazàn, and L. P. de las Heras, "Icdar 2013 robust reading competition," in *12th International Conference on Document Analysis and Recognition*, USA: IEEE Computer Society, 2013, pp. 1484–1493.

[43] K. Wang and S. J. Belongie, "Word spotting in the wild," in *Computer Vision – ECCV*, 2010.